MICROELECTRONIC SYSTEMS DESIGN RESEARCH GROUP

Workshop Session on:

3D Systems for Machine Learning and Memory architecture

Challenges of 3D DRAM Memories

Bandwidth, Power, Temperature, Reliability

Dr.-Ing. Christian Weis

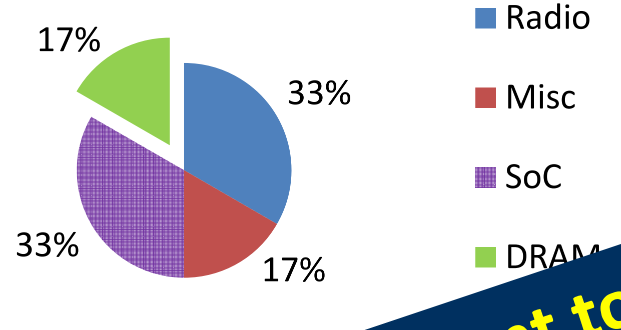
OPRECOMP Open Transprecision Computing

leti ceatech TECHNOLOGY RESEARCH INSTITUTE

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

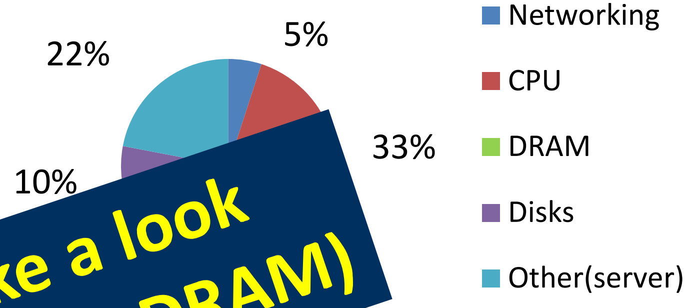
Why do we care about DRAM ?

Power Break-Down for Suspended 3G State



Source: The systems hacker... usage i...

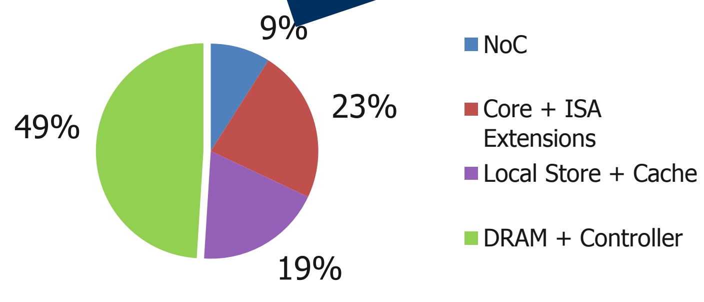
Power Break-Down Google Datacenter



...Datacenter as a Computer: An Introduction ...to the Design of Warehouse-Scale Machines. 2009

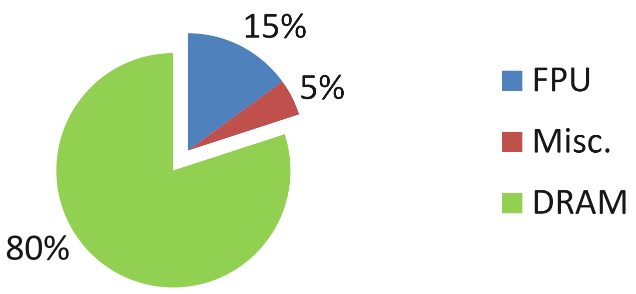
Important to take a look at the memories (e.g. DRAM)

Power B... Data... ...ion



Source: Power Consumption of Green Wave Architecture 2011

Power Break-Down eBrain

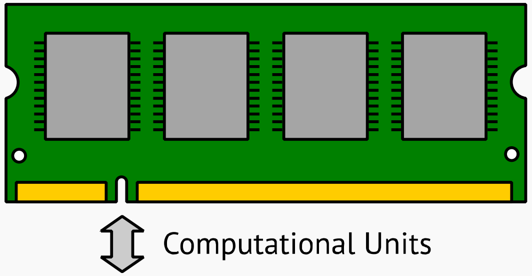


Source: A Scalable Custom Simulation Machine for the Bayesian Confidence Propagation Neural Network model of the Brain, 2014

Comparison of DRAM Subsystems

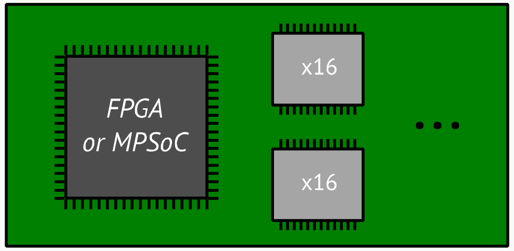
DIMM Based:

General Purpose Computers
e.g. DDR3, DDR4



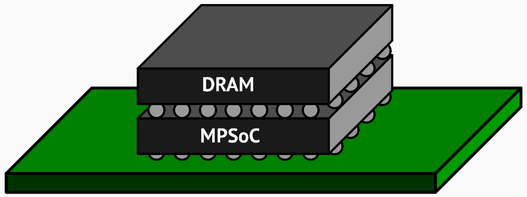
Device Based:

Embedded / Tablets / Graphic Cards
e.g. LPDDR3, GDDR5



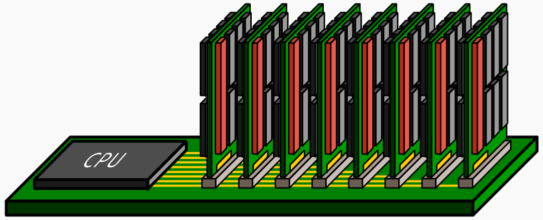
Package on Package (PoP):

Soldered on top of the MPSoC.
Smartphones
e.g. LPDDR3, LPDDR4



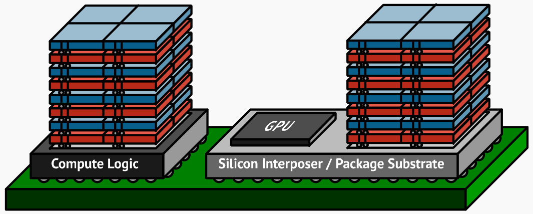
Buffer on Board:

Memory Controller on Buffer Chip,
Serial Connection
e.g. FBDIMM, IBM CDIMM, Intel SMI/SMB



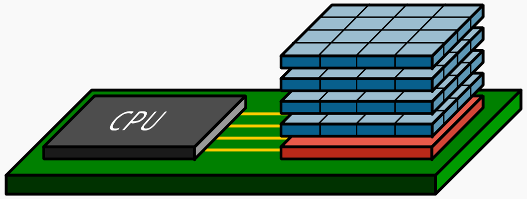
3D/2.5D-Integrated:

Stacked on Logic or Silicon Interposer
by means of TSVs
e.g. Wide I/O, HBM



Memory Cube:

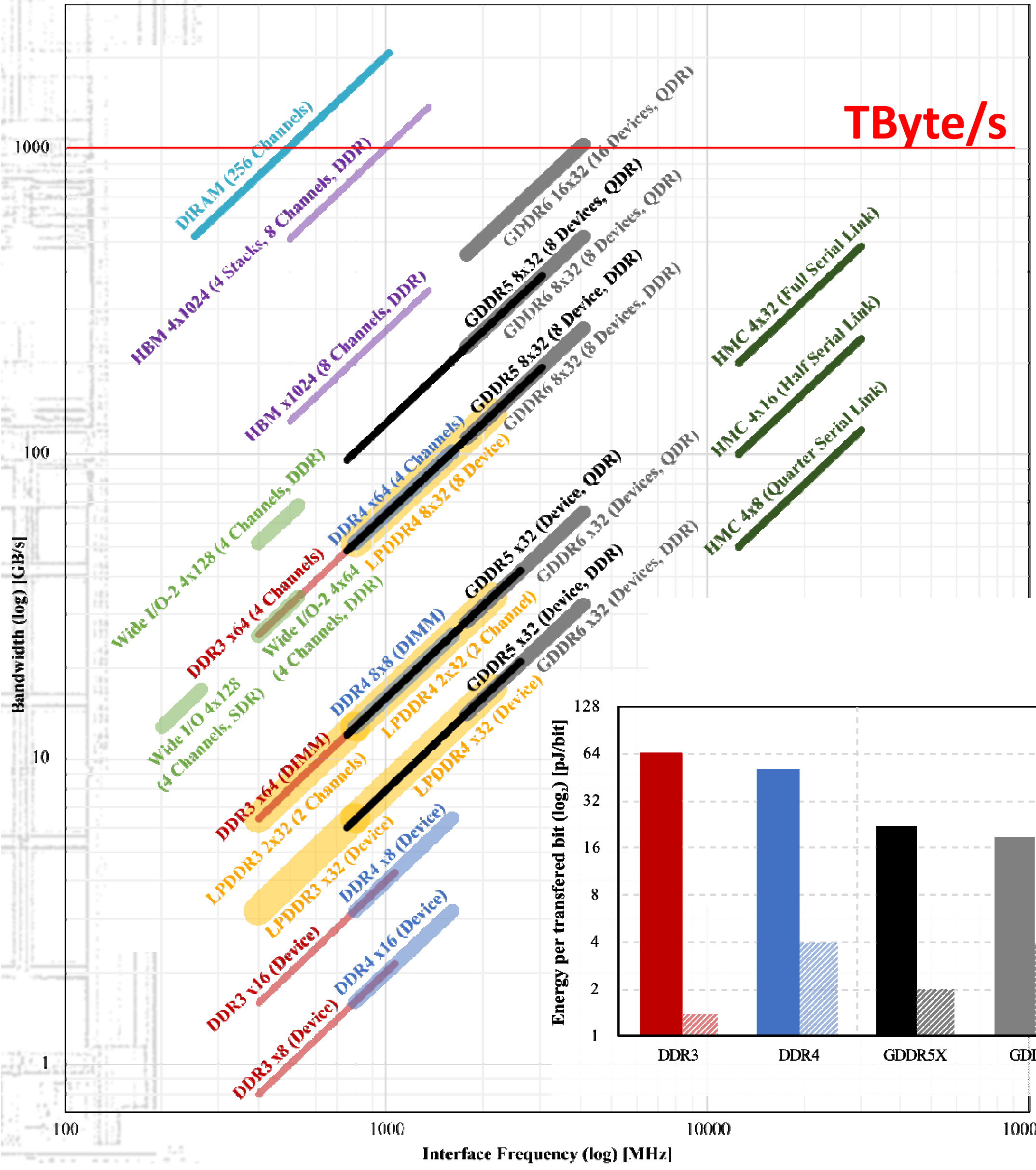
3D-Stacked, Memory Controller on
Bottom Layer, Serial Interconnect (SerDes)
e.g. HMC, SMC

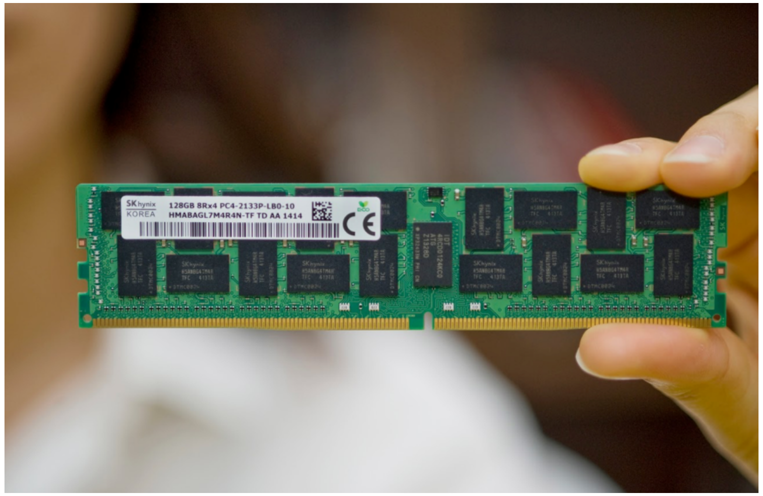


Source: Matthias Jung

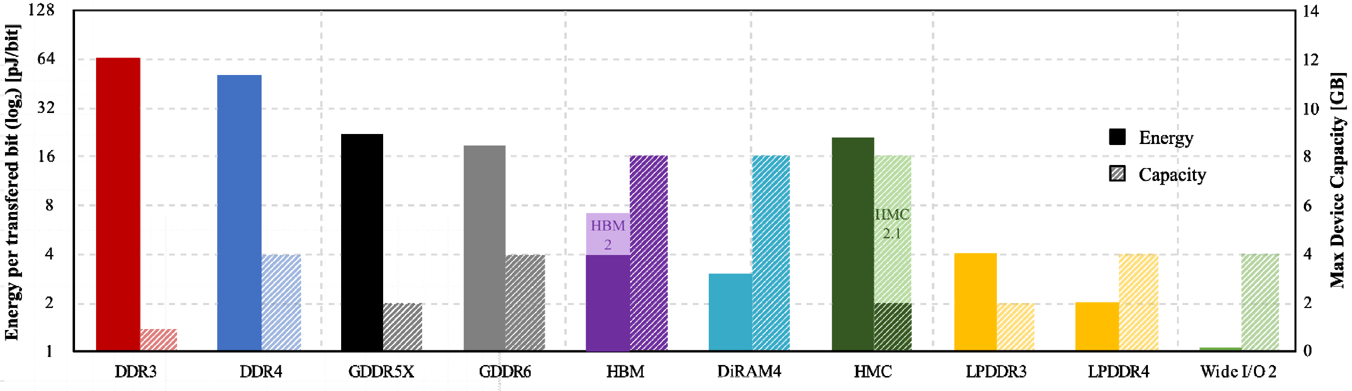
MICROELECTRONIC SYSTEMS DESIGN RESEARCH GROUP

Comparison of DRAM Subsystems





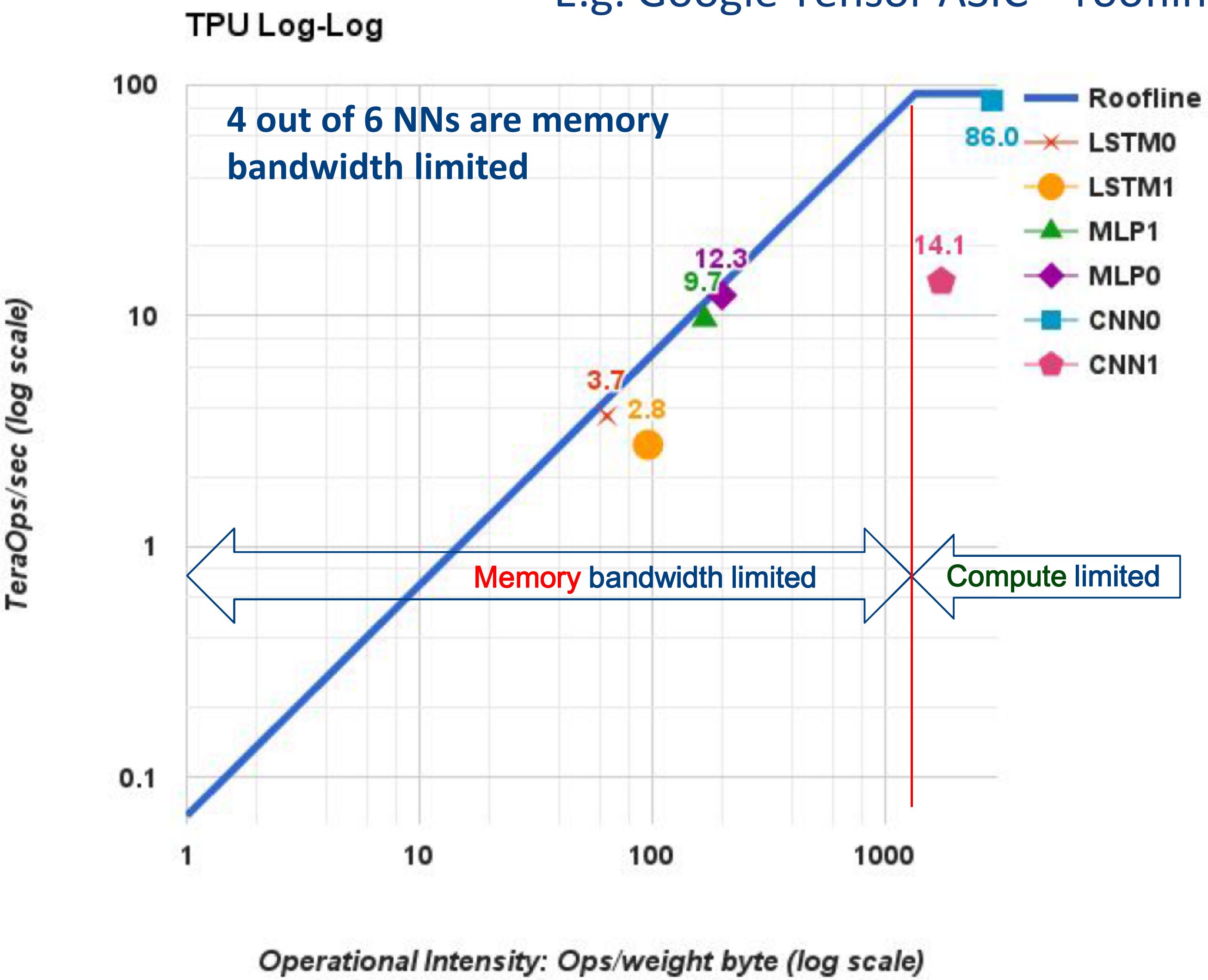
Best case -
100% usage of the available BW



Source: Matthias Jung

Bandwidth Challenge

E.g. Google Tensor ASIC – roofline Model

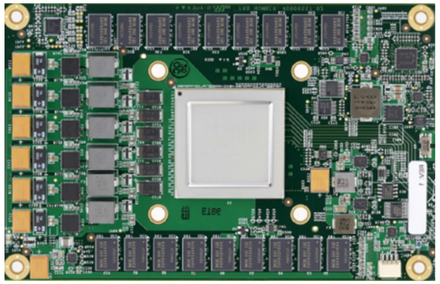


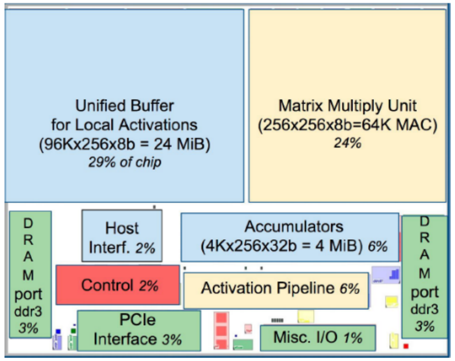
4 out of 6 NNs are memory bandwidth limited

Memory bandwidth limited

Compute limited

64K MAC Units
8 GByte DDR3





Unified Buffer for Local Activations (96Kx256x8b = 24 MiB) *29% of chip*

Matrix Multiply Unit (256x256x8b=64K MAC) *24%*

DRAM port ddr3 *3%*

Host Interf. *2%*

Accumulators (4Kx256x32b = 4 MiB) *6%*

DRAM port ddr3 *3%*

Control *2%*

Activation Pipeline *6%*

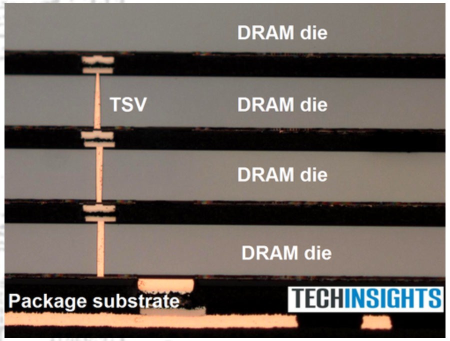
PCIe Interface *3%*

Misc. I/O *1%*

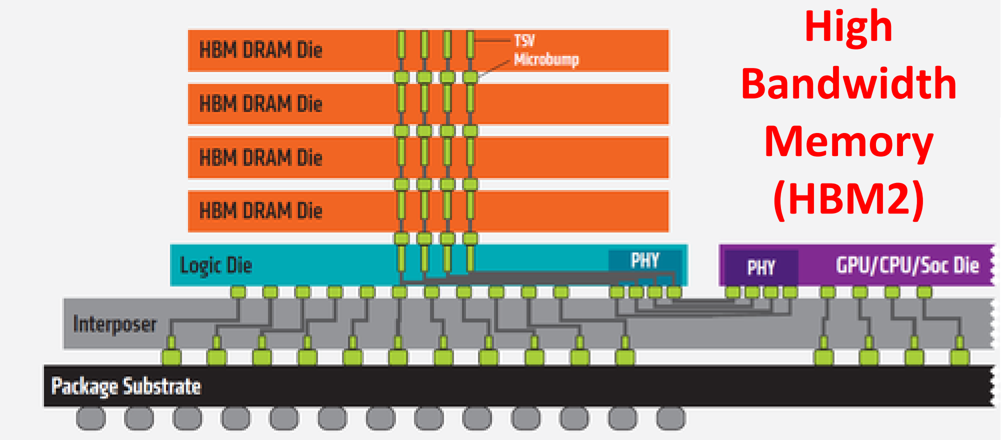
Bandwidth Challenge in 3D-DRAMs

HBM2 and HMC can provide huge bandwidths.

BUT, there is a price to pay …

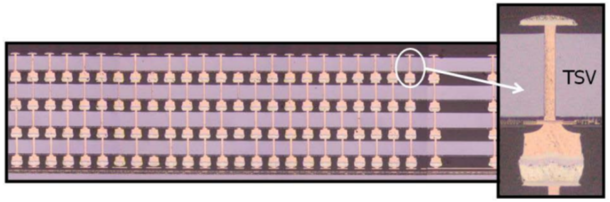


>256 GB/s
Bandwidth / Cube



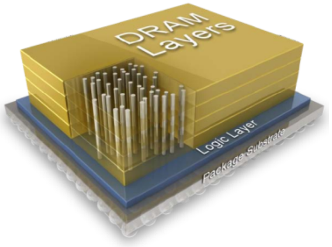
High Bandwidth Memory (HBM2)

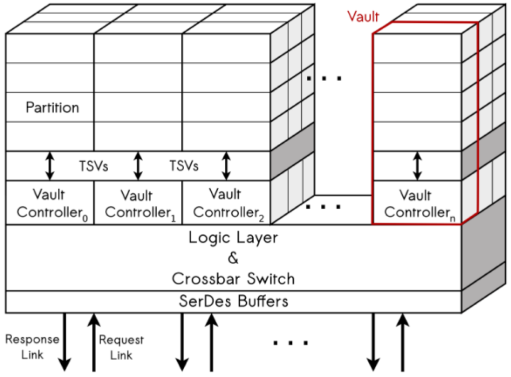
Power: 40-50 GB/s/W / Cube



Hybrid Memory Cube (HMC)

>240 GB/s
Bandwidth / Cube





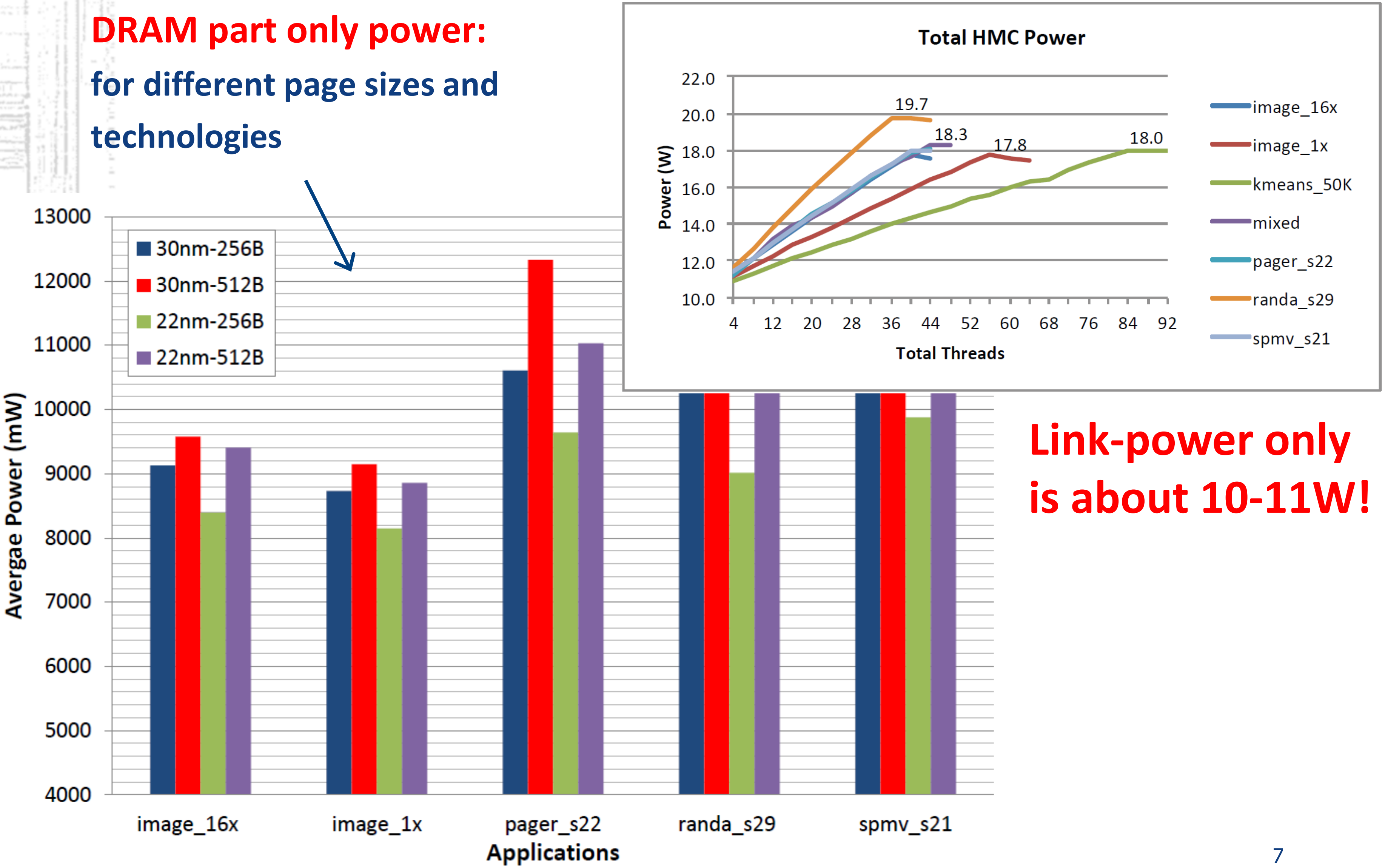
Power: ≈20 W / Cube

HMC Power >>11W

M. Gokhale et al., 2015

DRAM part only power:

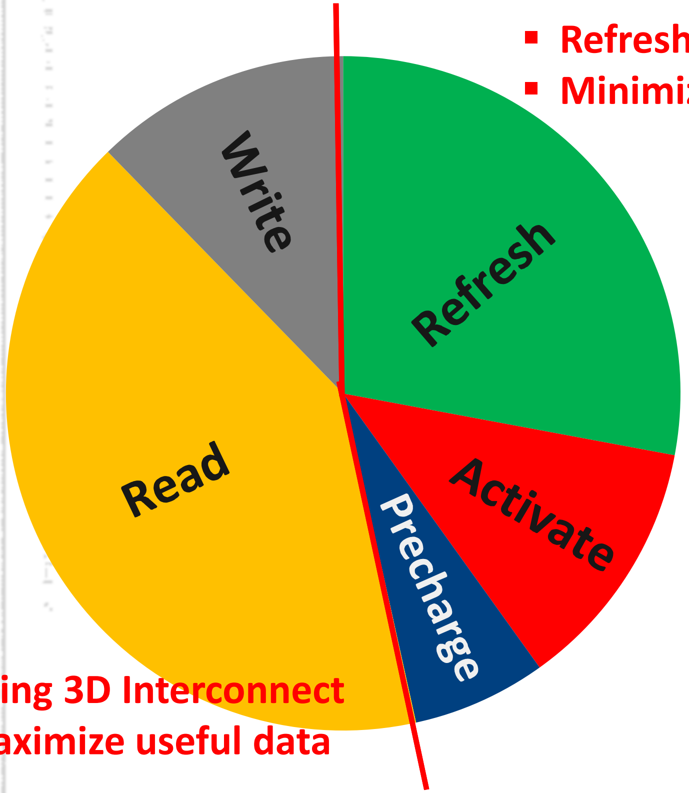
for different page sizes and technologies



Link-power only is about 10-11W!

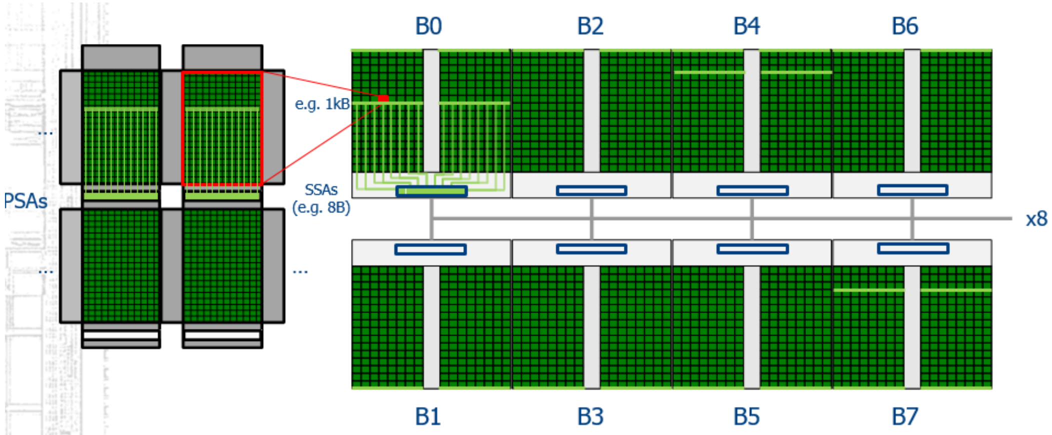
Detailed DRAM Energy Distribution

- DRAM Power Breakdown for Twitter Memcached Application*
- 2GB LPDDR3 (Low-Power DDR3 DRAM)



- **Refresh optimizations**
- **Minimizing Row misses**

- **Using 3D Interconnect**
- **Maximize useful data**



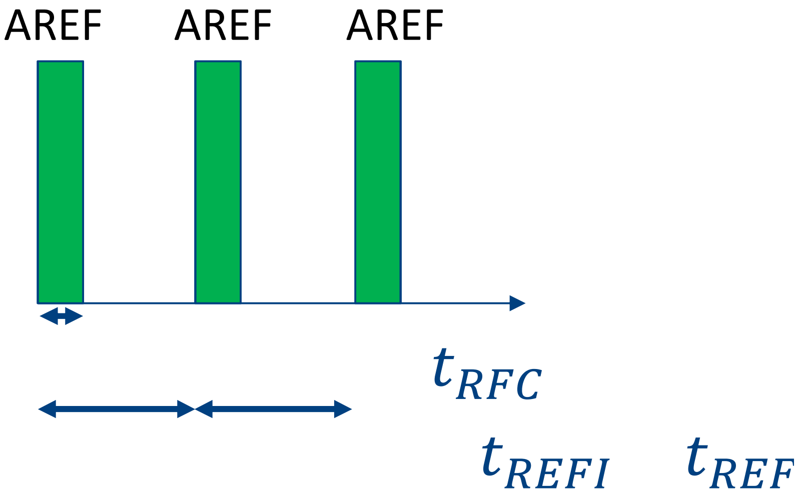
Important DRAM Commands

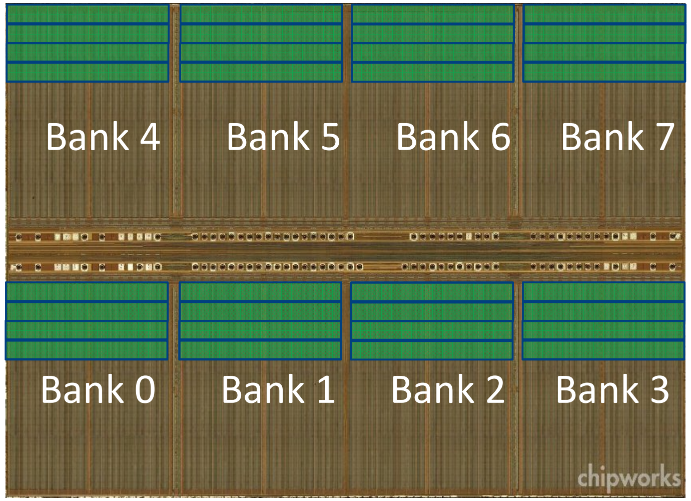
ACT: Activates a specific row in a specific bank (sensing into PSA)	*tRCD*
RD: Read from activated row (prefetch from PSA to SSA and burst out)	*tCL + tBURST*
PRE: Precharges set LWL=0 set LBL=VDD/2	*tRP*
REFA: DRAM cells are leaky and have to be refreshed	*tREFI & tREF*

(*) A High-Level DRAM Timing, Power and Area Exploration Tool, O. Naji, A. Hansson, C. Weis, M. Jung, N. Wehn *IEEE International Conference on Embedded Computer Systems Architectures Modeling and Simulation (SAMOS)*, July 2015

How Refresh is Performed?

- DRAM controller sends AREF commands every t_{REFI} (eg. 7.8 µs for Temp. < 85°C)
- Single AREF command refreshes multiple rows in all banks (eg. '2' rows in all 8 banks for 2Gb DDR3 DRAM)



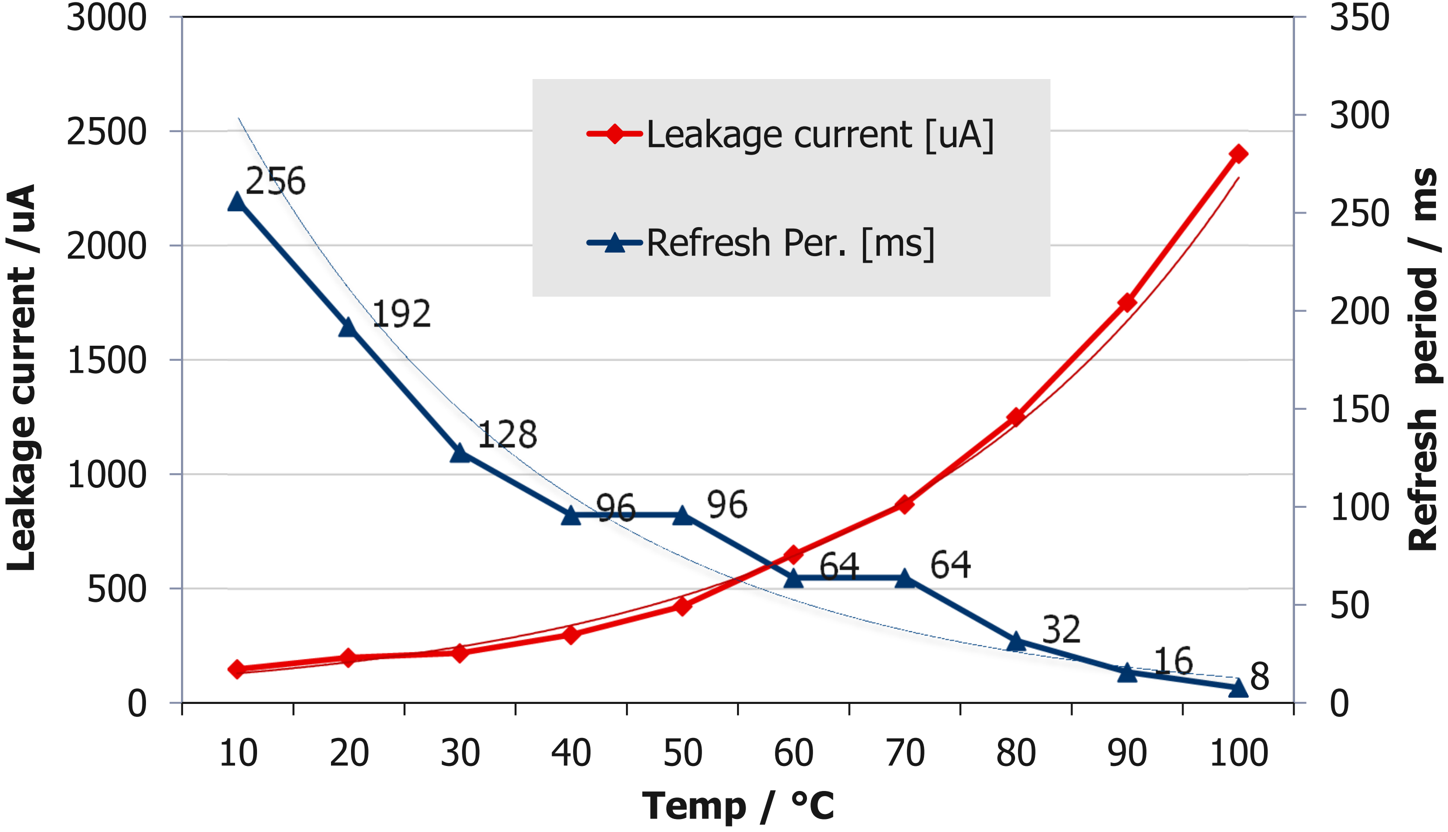


MICROELECTRONIC
SYSTEMS DESIGN
RESEARCH GROUP

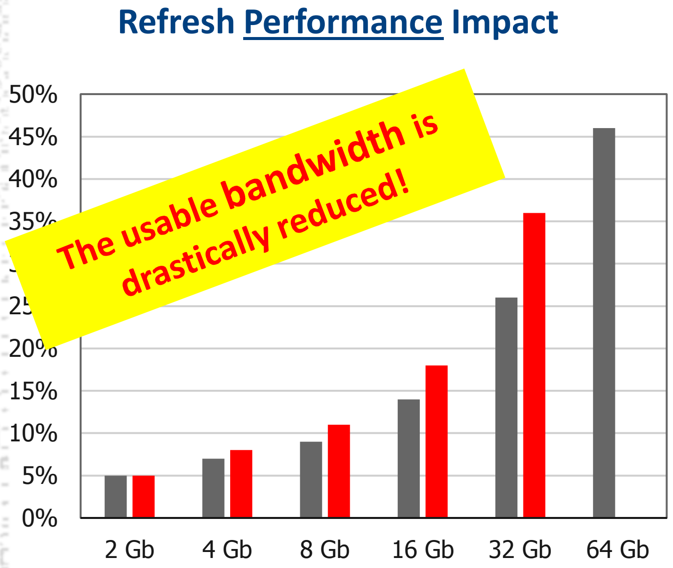
Refresh/Temperature Challenge

Exponential temperature/leakage current behavior
→ shorter refresh periods

LPDDR3 Datasheet

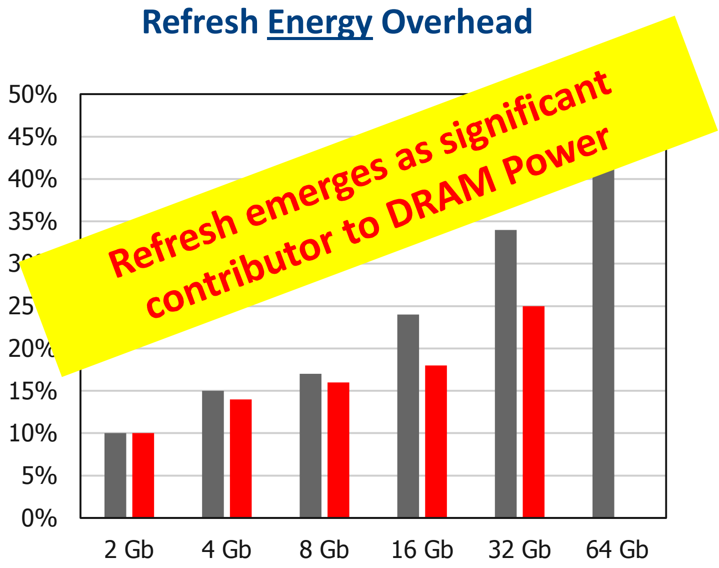


Refresh/Temperature Challenge



Refresh Performance Impact

The usable bandwidth is drastically reduced!

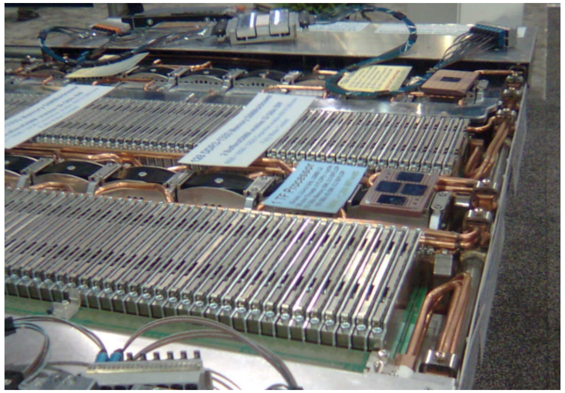


Refresh Energy Overhead

Refresh emerges as significant contributor to DRAM Power

⬛ *J. Liu, et al. RAIDR: Retention-Aware Intelligent DRAM Refresh, ISCA 2012*

🟥 *I. Bhati, et al. DRAM Refresh Mechanisms, Trade-offs and Penalties, IEEE Trans. 2015*



4 TB DDR3 DRAM
Stand-by 300W (only Refresh)

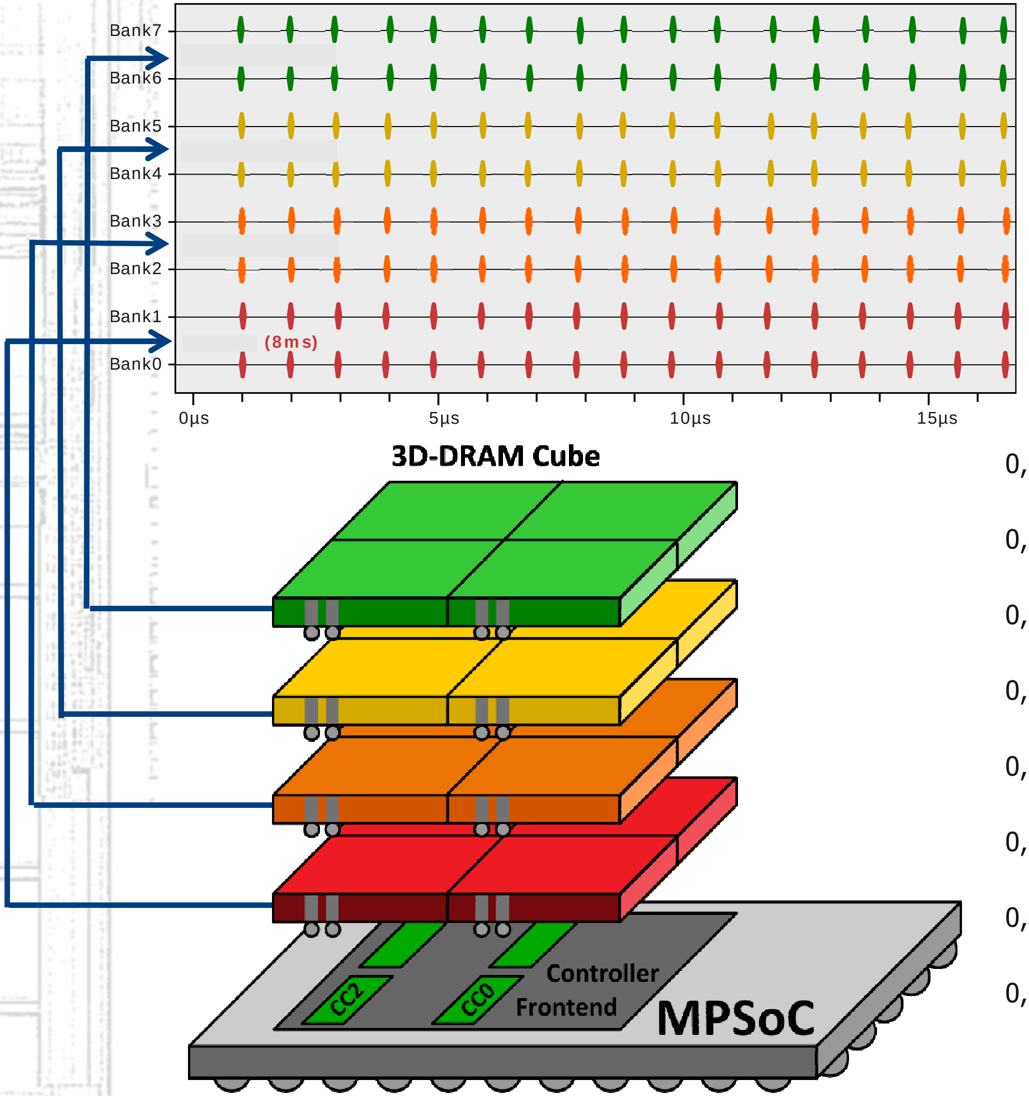
Paul Rosenfeld (IBM Server on display at Supercomputing)

Refresh Optimization Techniques

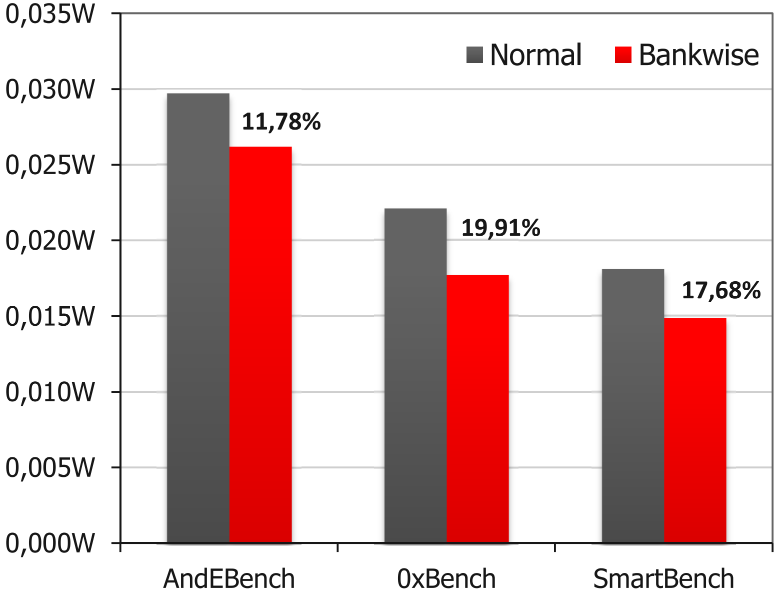
To counterbalance this trend for future devices and the higher temperatures in 3D-DRAMs …

- **Temperature-aware Bank-wise** Refresh (detailed control)
- **Approximate** DRAM
- **ORGR** – Optimized Row Granular Refresh (only refresh data that is stored in an optimized way)

Temperature-aware Bank-Wise Refresh



- Different refresh rates on different dies (bank groups), according to the temperature of the die/bank
- Each bank was equipped with a TS (Temp Sensor)

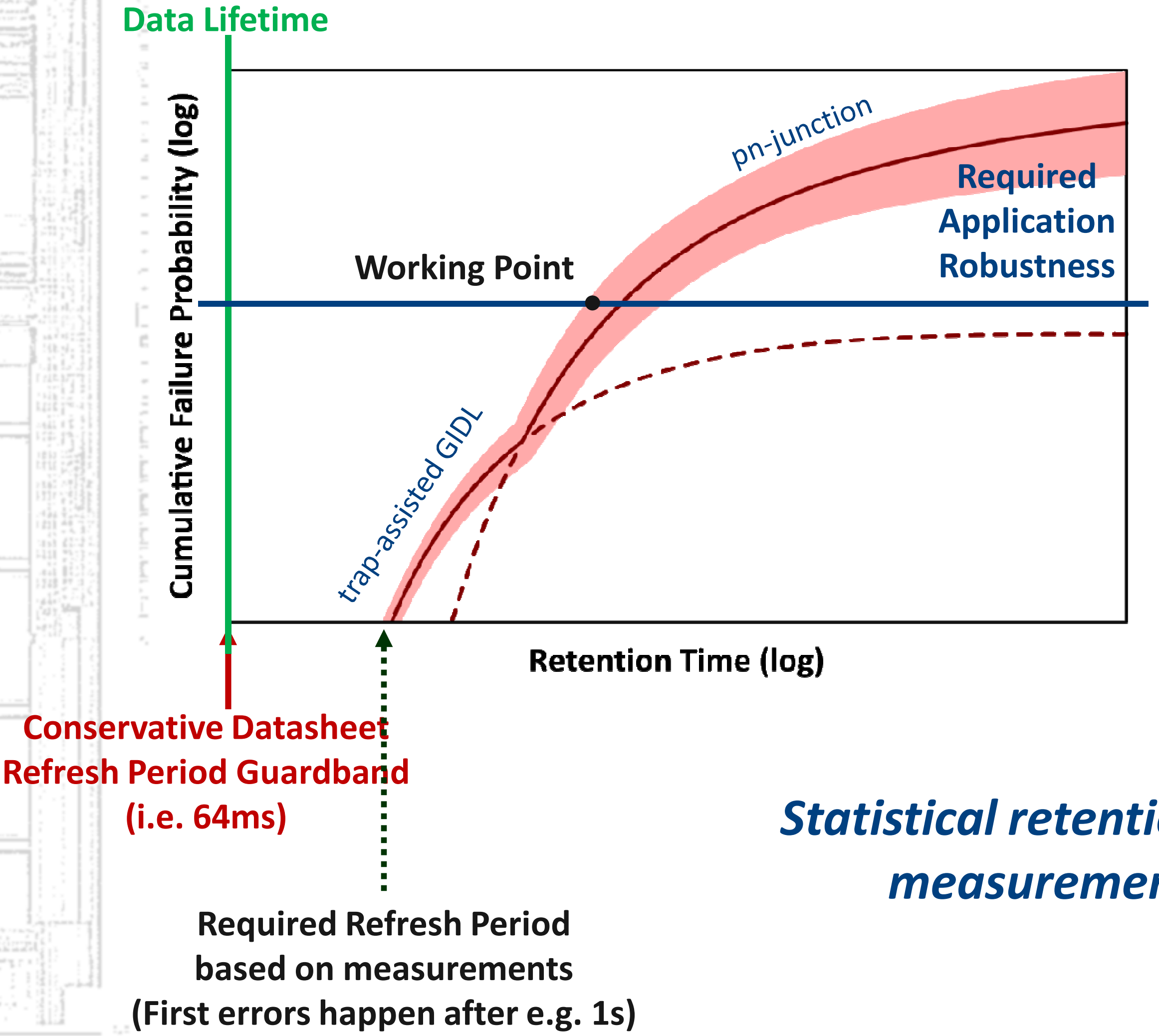


M. Sadri, et al. Energy Optimization in 3D MPSoCs with Wide-I/O DRAM Using Temperature Variation Aware Bank-wise Refresh, DATE 2014

Approximate DRAM

Lowering or completely switching off refresh, accepting risk of data errors

- Consider DRAM device as a stochastic model that includes process variations



Conservative Datasheet Refresh Period Guardband (i.e. 64ms)

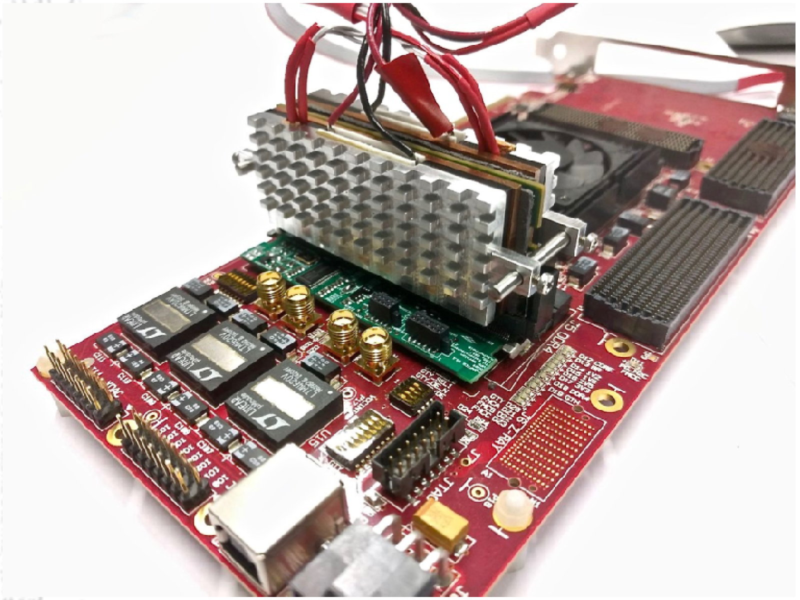
Required Refresh Period based on measurements (First errors happen after e.g. 1s)

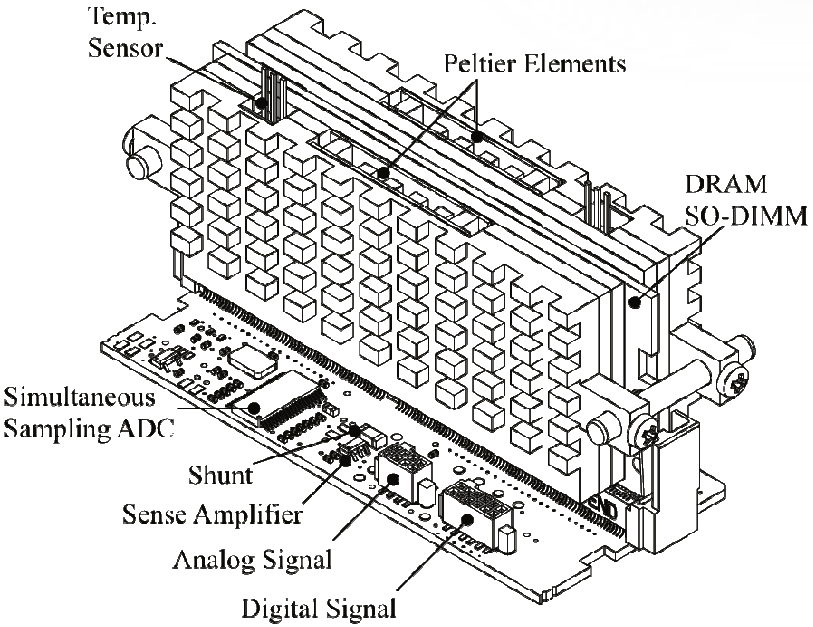
Switch Off Refresh

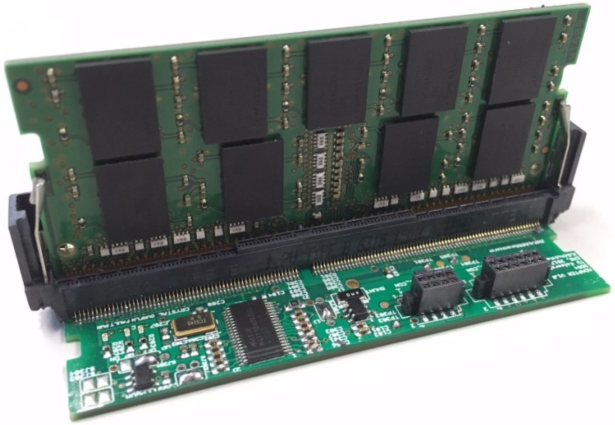
- If data lifetime is smaller than required refresh period
- If data lifetime is larger than required refresh period AND application has some robustness w.r.t. errors

Statistical retention error model and measurements mandatory

DRAMMeasure DDR4



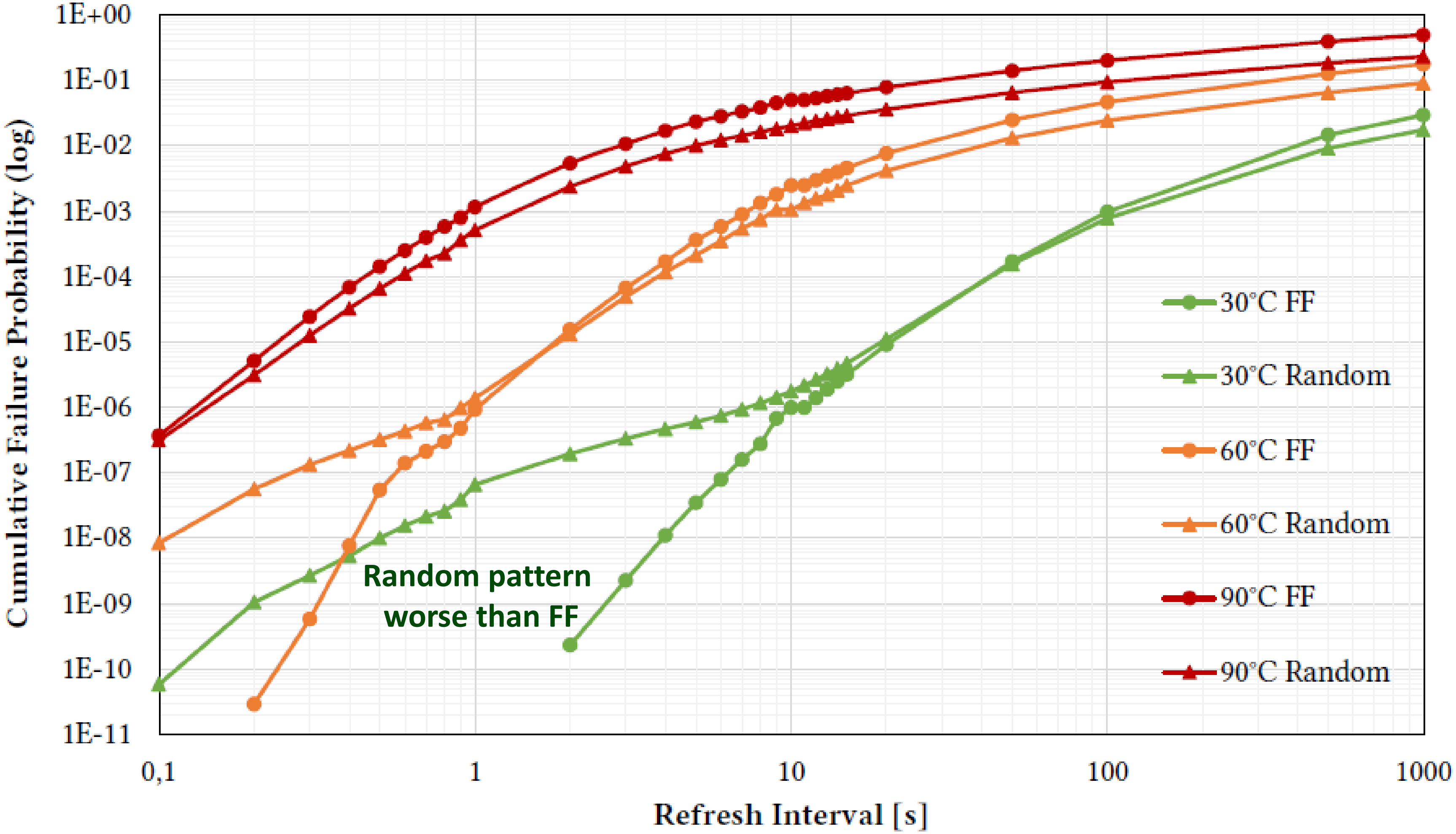




- High freq. 1.2GHz and higher: DDR4-2400
- Precise temperatures for heating up to 95°C
- Exact current measurements incl. VPP

DDR4 Retention Time Measurements I

- Retention behavior depends on cell leakage (drain, sub-threshold, cell capacitor), cross talk, process variations, temperature, cell type

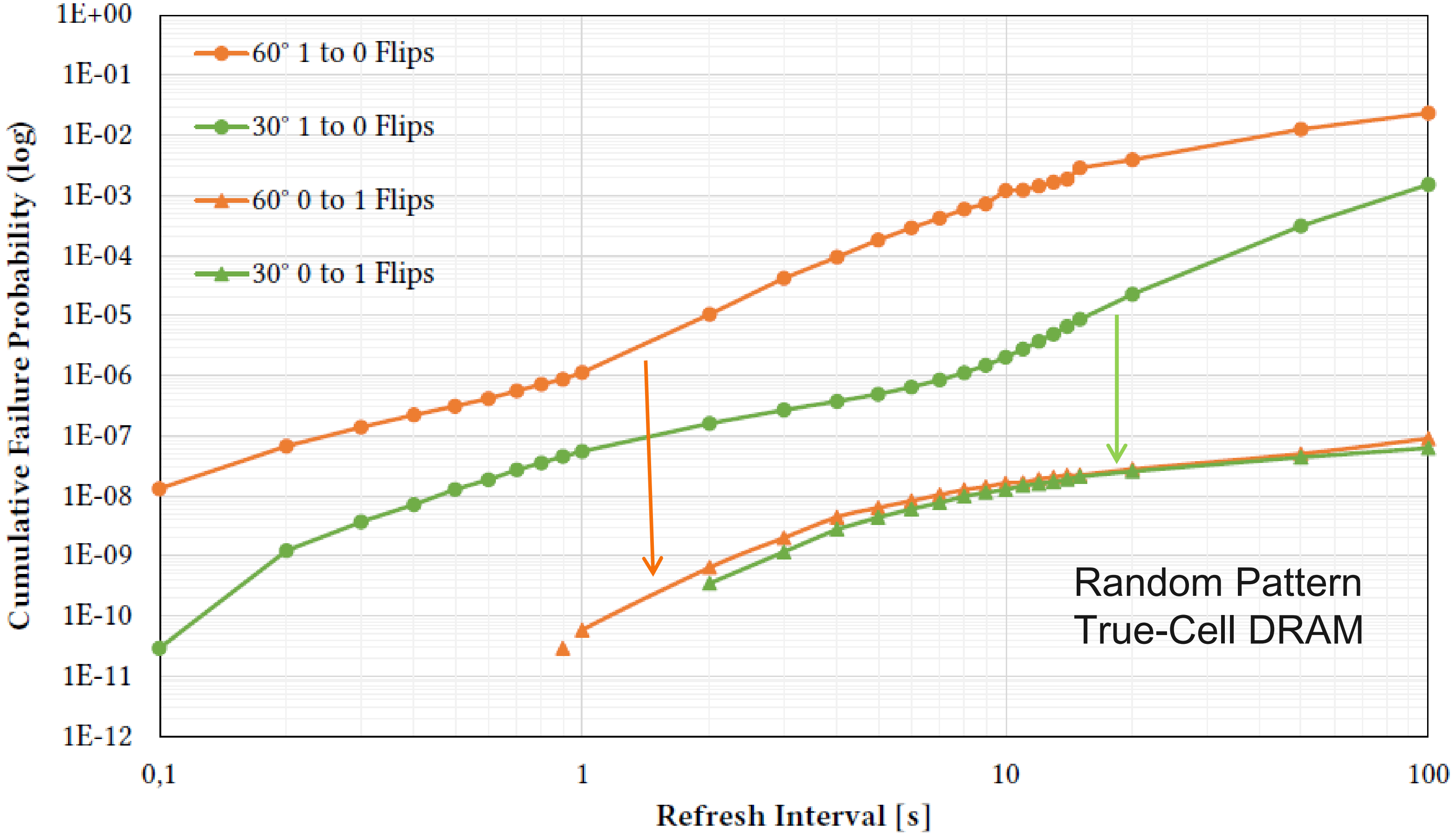


MICROELECTRONIC SYSTEMS DESIGN RESEARCH GROUP

DRAMMeasure

DDR4 Retention Time Measurements II

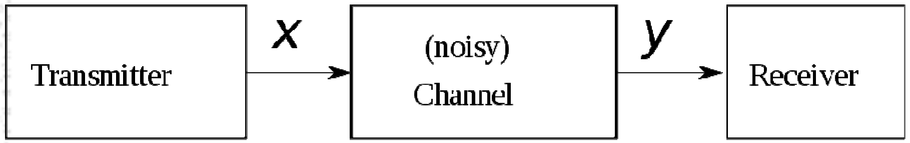
- Unsymmetrical error behavior dependent on cell type (true-cell, anti-cell)



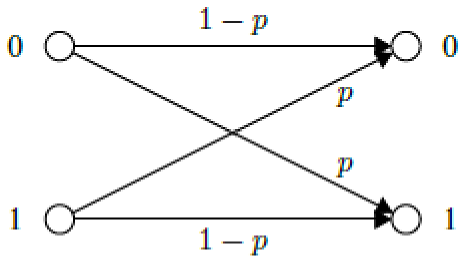
1→0 flip much more likely than 0→1 flip

Information Theory

Consider memory as noisy communication channel

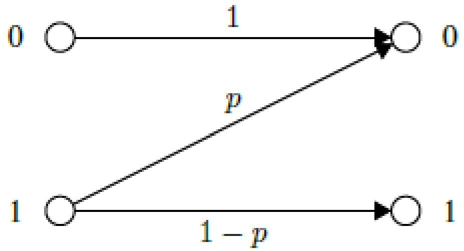


Symmetric retention behavior:
Binary Symmetric Channel (BSC)



$$C_{BSC} = 1 - H(p)$$

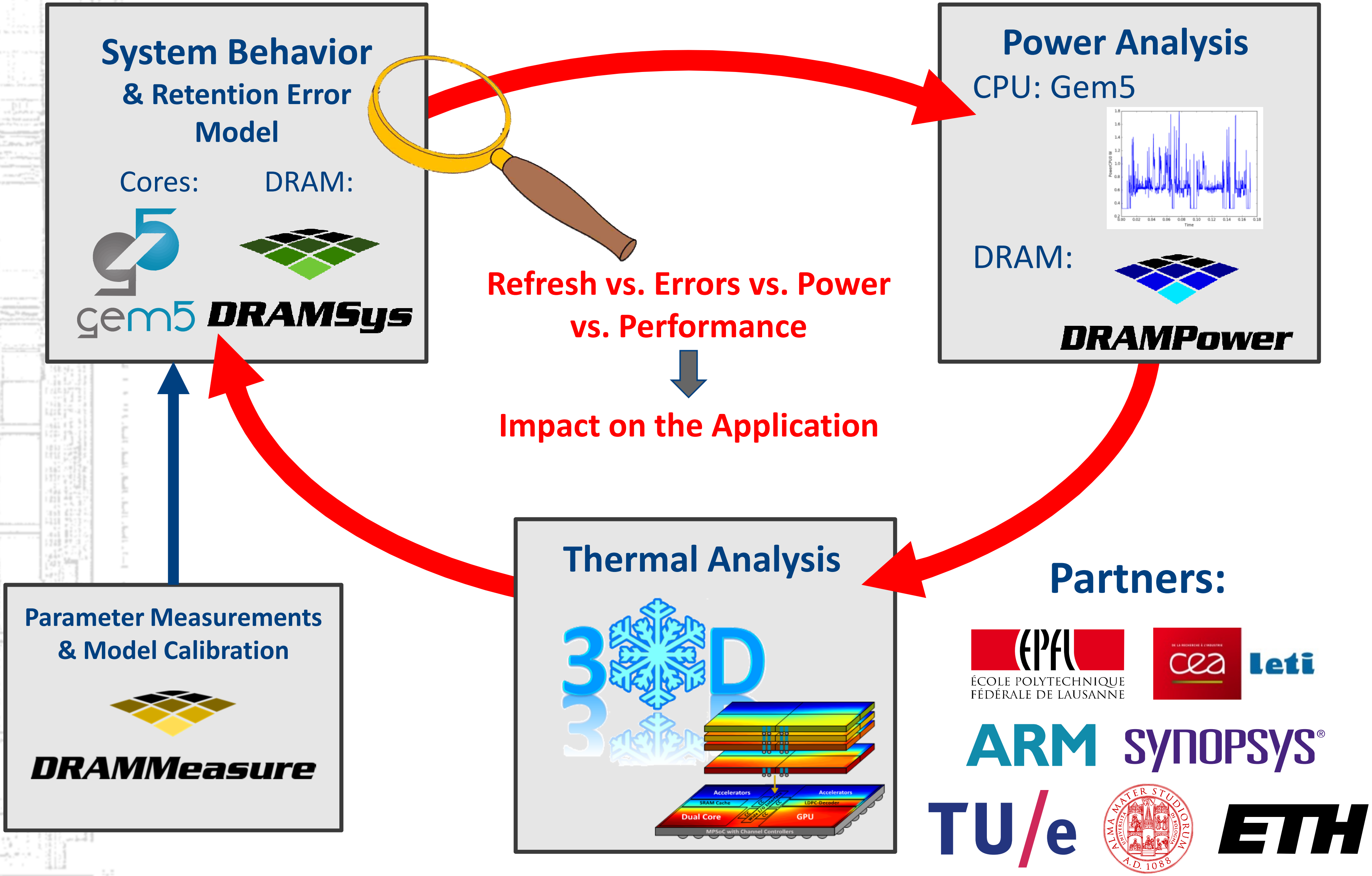
Asymmetric retention behavior:
Z-Channel



$$C_Z = \log_2\left(1 + (1-p)\cdot p^{\frac{p}{1-p}}\right) \approx 1 - \frac{1}{2}H(p)$$

- Larger reliability if internal cell structure is known
- More efficient **ECC** techniques possible
- Appropriate data representation: e.g. small dynamic range C2 versus sign/magnitude

Approximate DRAM Simulation Framework



System Behavior & Retention Error Model

Cores: gem5

DRAM: DRAMSys

Power Analysis

CPU: Gem5

DRAM: DRAMPower

Thermal Analysis

3D

Parameter Measurements & Model Calibration

DRAMMeasure

Refresh vs. Errors vs. Power vs. Performance

Impact on the Application

Partners:

A Per Layer Refresh Policy for 3D DRAMs

Separation of a 3D DRAM Stack into **unreliable** and **reliable** regions

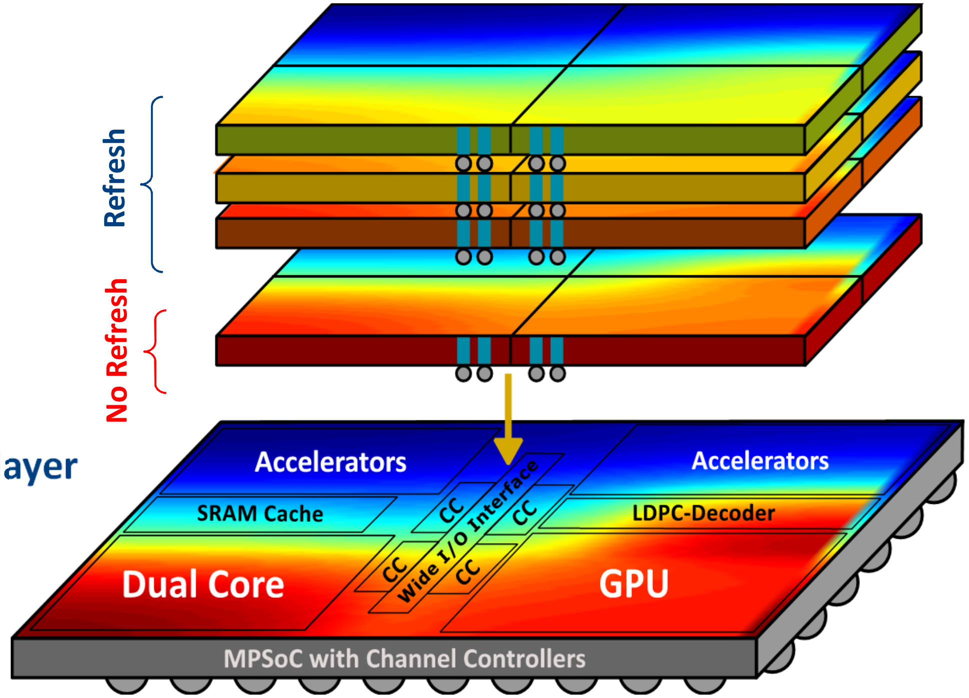
- Reliable regions: higher DRAM layers with temperature aware refresh
- Unreliable region: bottom DRAM layer with disabled refresh → **Omit Refresh (OR)**
- Access unreliable region while reliable region is refreshed

Typical example applications

- Graph processing
- Image processing
- Baseband processing

→ Saves 100% refresh power in the unr.-layer

→ Increases bandwidth



Simulation Results

Matthias Jung, et al. 2015. Omitting Refresh: A Case Study for Commodity and Wide I/O DRAMs.

Reducing Refresh Overhead

- **Selective Refresh**
- **Retention Aware Refresh**
- **Approximate DRAM**

**Different rows need to be refreshed
at different rates**

Drawbacks of normal Auto-Refresh:

- AREF lacks flexibility
- No access to internal refresh row counter
- No rows can be skipped
- The complete DRAM has to be refreshed in the same rate

Use

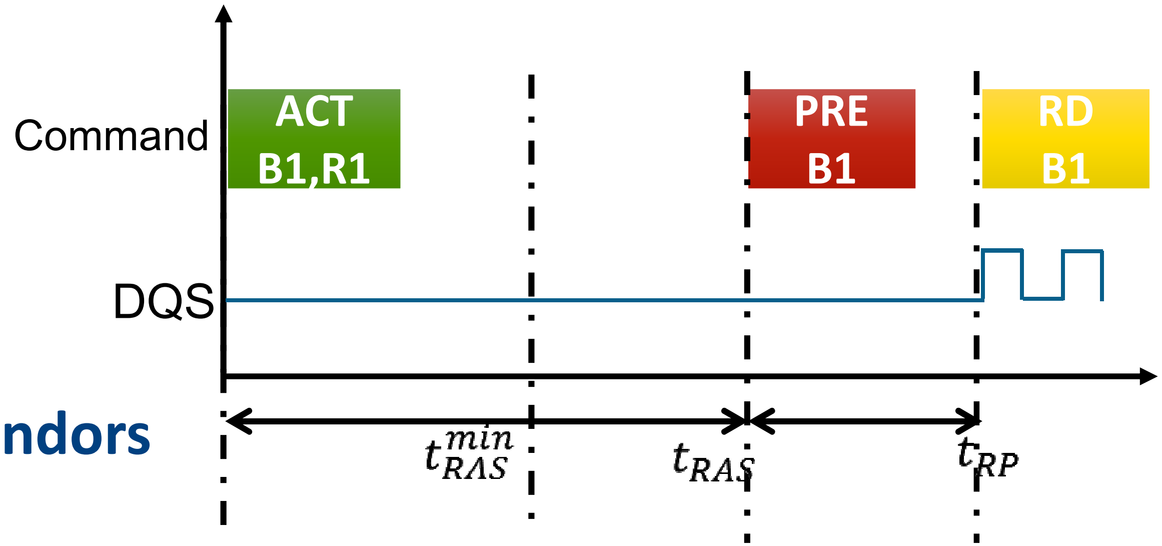
Optimized

Row

Granular

Refresh

ORGR – Idea / Vendor Specific



- t_{RASmin} timer at all vendors present (safety)!
- But, vendor specific implementation
- Reverse Engineering technique performed during init, boot or run-time

t_{RAS} = 37.5 ns

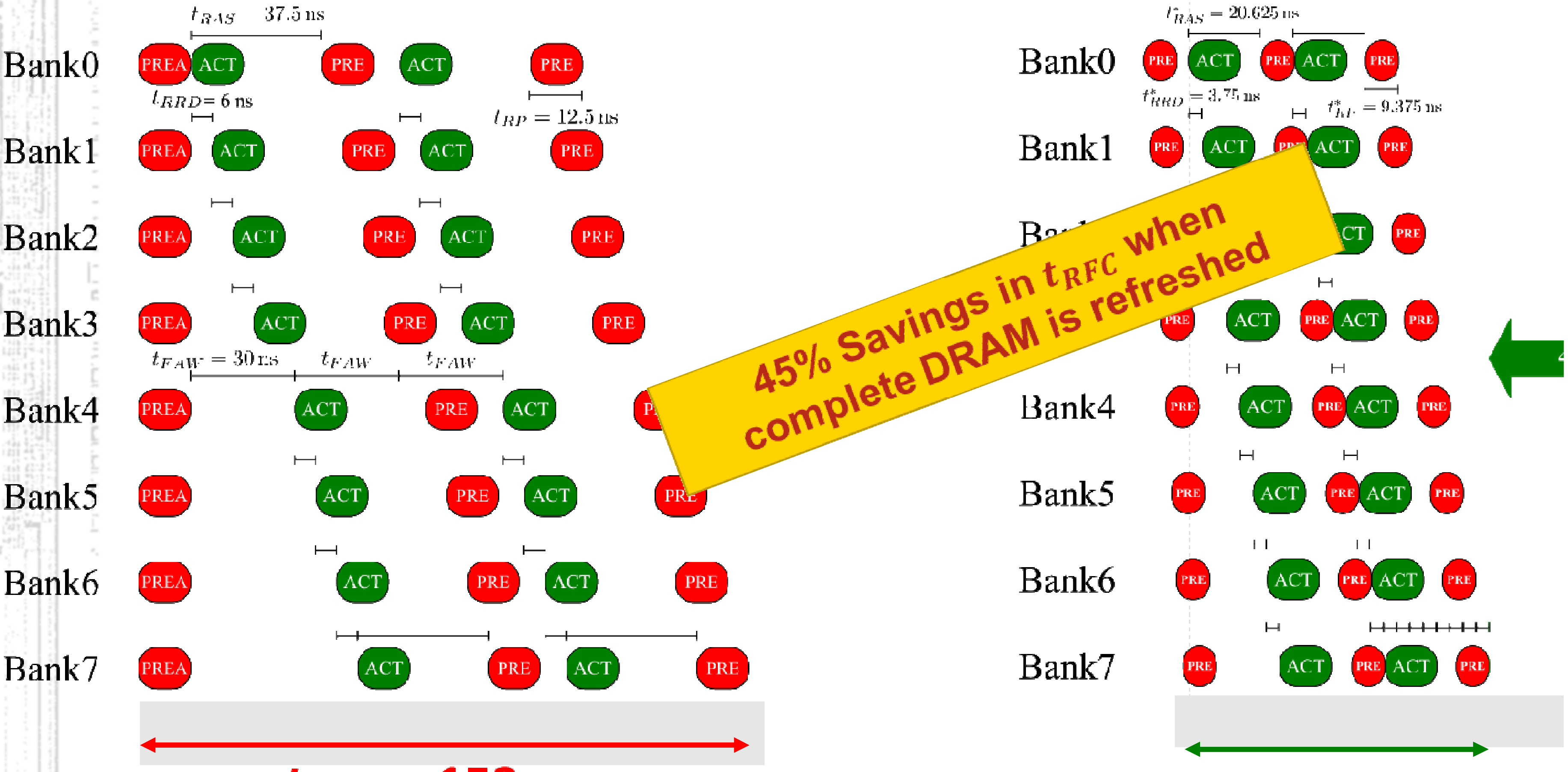
t_{RAS}^{min} = 20.7 ns

Patent Pending

ORGR - Benefits

RGR (based on JEDEC timings)

ORGR



45% Savings in t_{RFC} when complete DRAM is refreshed

t_{RFC} = 158 ns

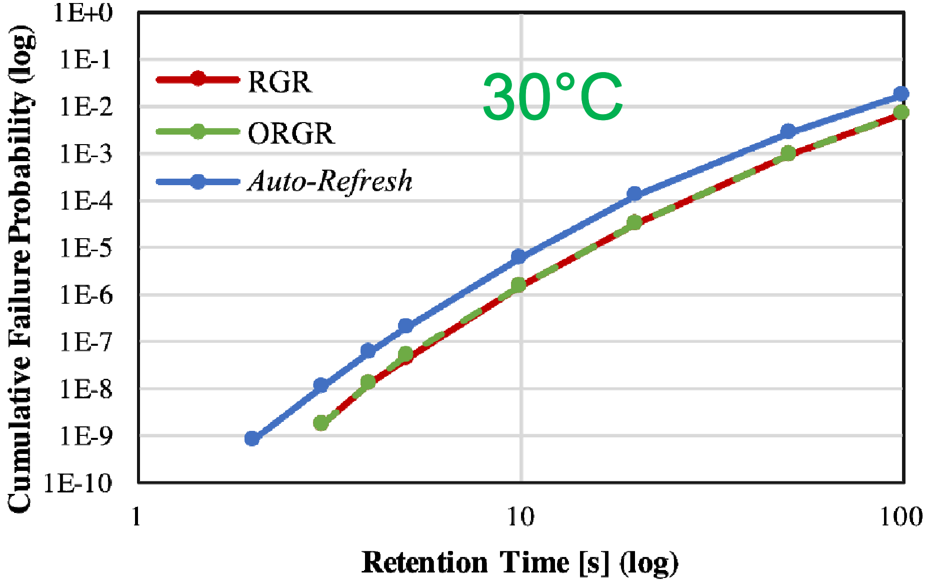
Patent Pending

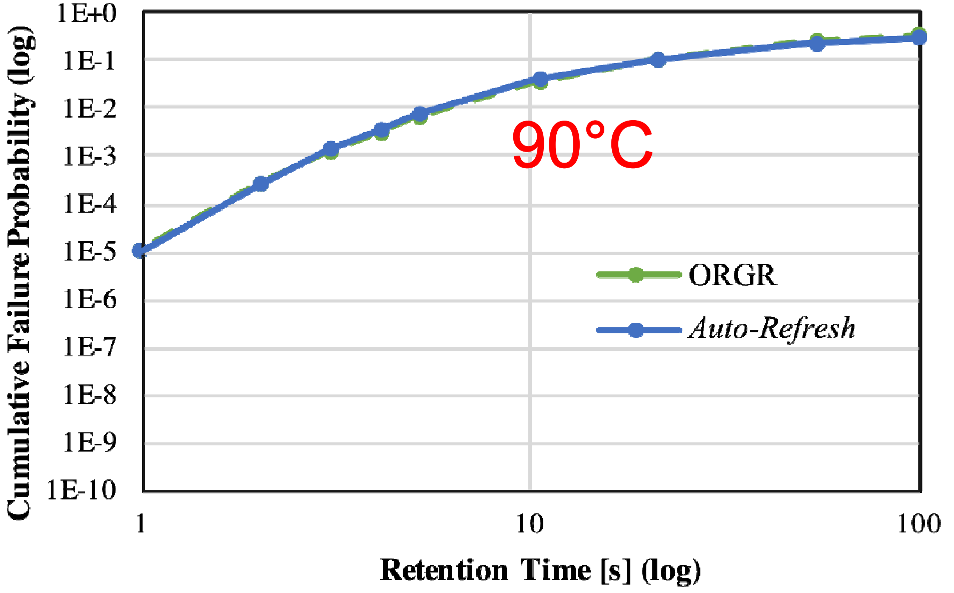
t_{RFC} = 86.25 ns

ORGR – Validation/Reliability

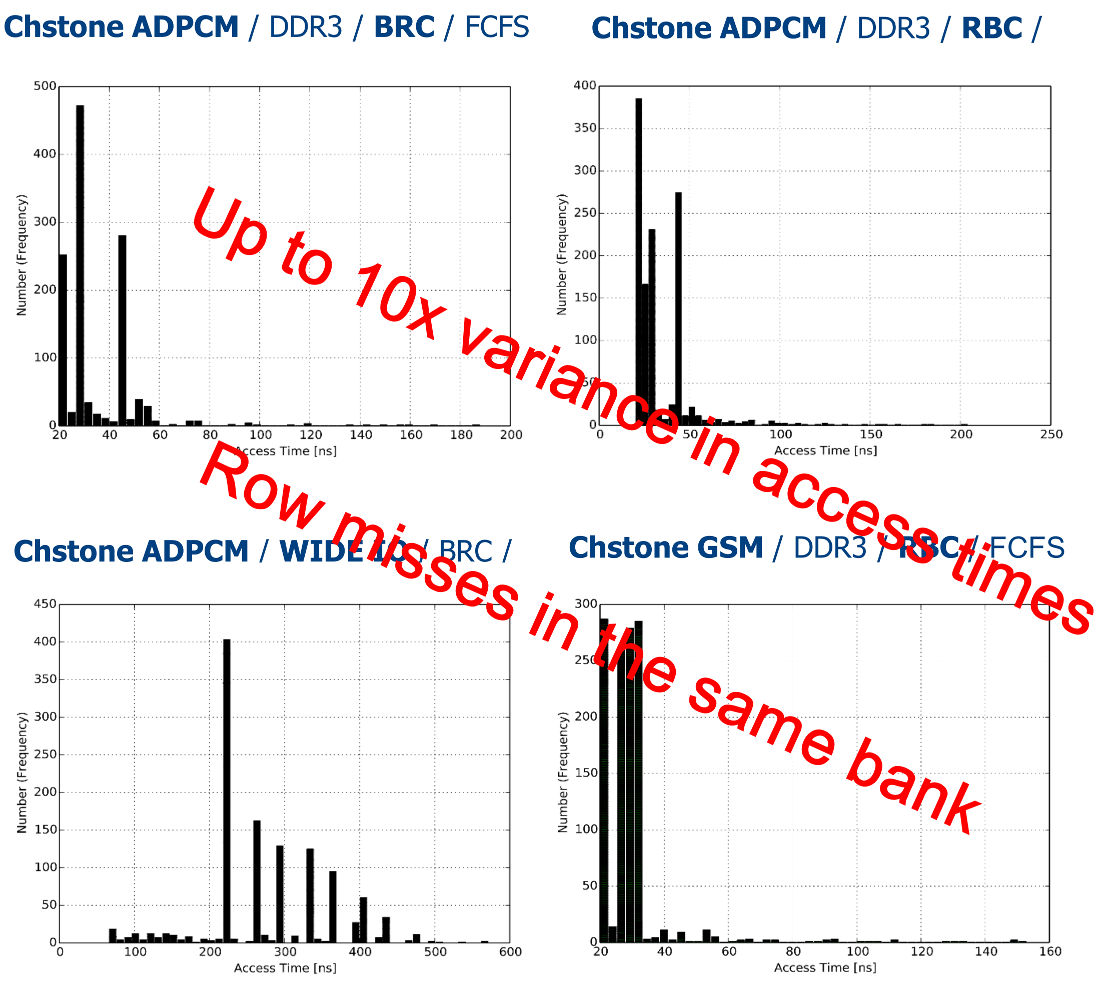
Refresh Technique	t_{RFC} /ns	Refresh Energy /mJ
Auto Refresh	262.5	186.24
RGR	292.5	230.48
ORGR	146.25	209.72

- Measured for 4Gb x16 DDR3 DRAM
- Refreshing the complete DRAM





“Non-deterministic” DRAM Timing Behavior



Chstone ADPCM / DDR3 / **BRC** / FCFS

Chstone ADPCM / DDR3 / **RBC** /

Chstone ADPCM / **WIDE IO** / BRC /

Chstone GSM / DDR3 / **RBC** / FCFS

Up to 10x variance in access times

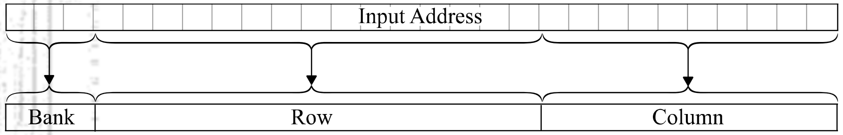
Row misses in the same bank

- DRAM latency varies largely
- Depending on
  - **Application**
  - **Address Mapping**
  - **DRAM**
  - **Memory Controller**
  - …

- Similar variation in energy/DRAM access

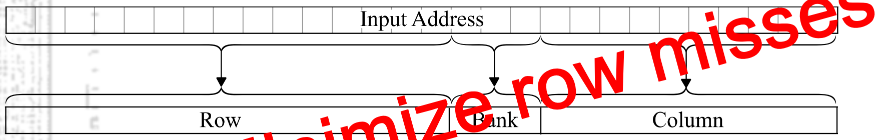
Application Aware Address Mapping

Standard Mapping (BRC)


Input Address

Bank Row Column

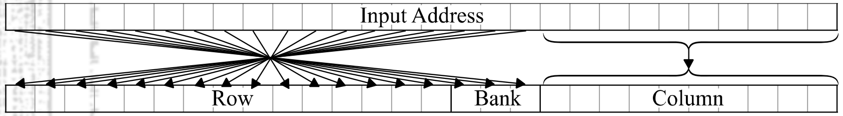
Bank Interleaving (RBC)


Input Address

Row Bank Column

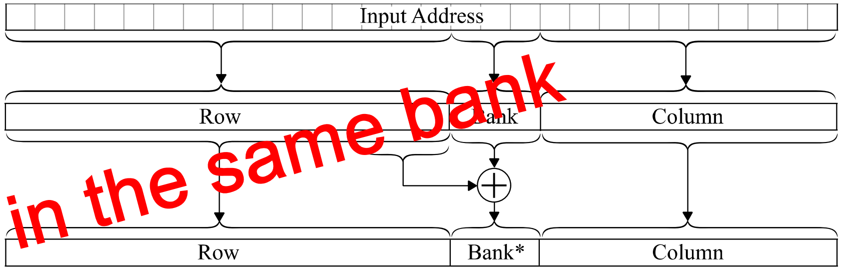
Minimize row misses in the same bank

Bit Reversal Address Mapping


Input Address

Row Bank Column

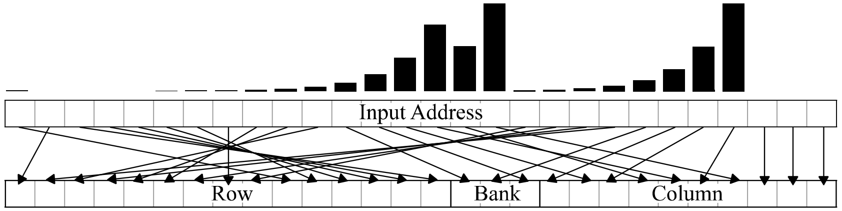
Permutation-Based Page Interleaving


Input Address

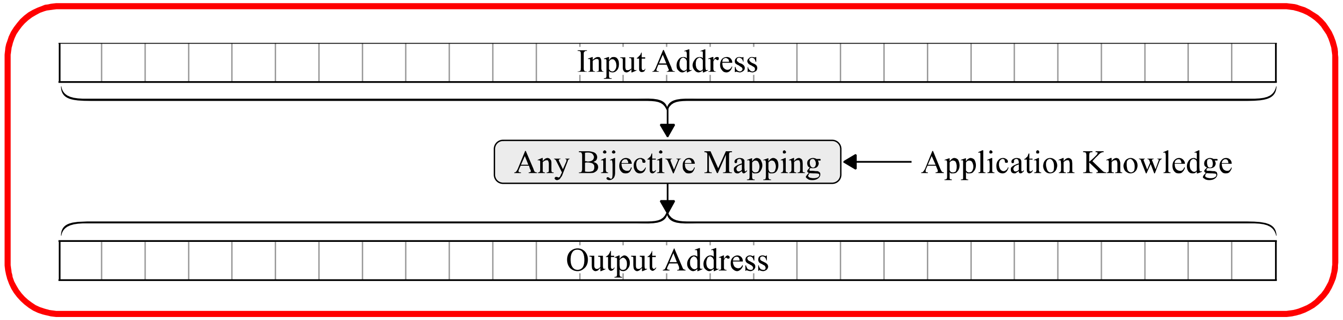
Row Bank Column

Row Bank* Column

Toggling Rate Analysis


Input Address

Row Bank Column


Input Address

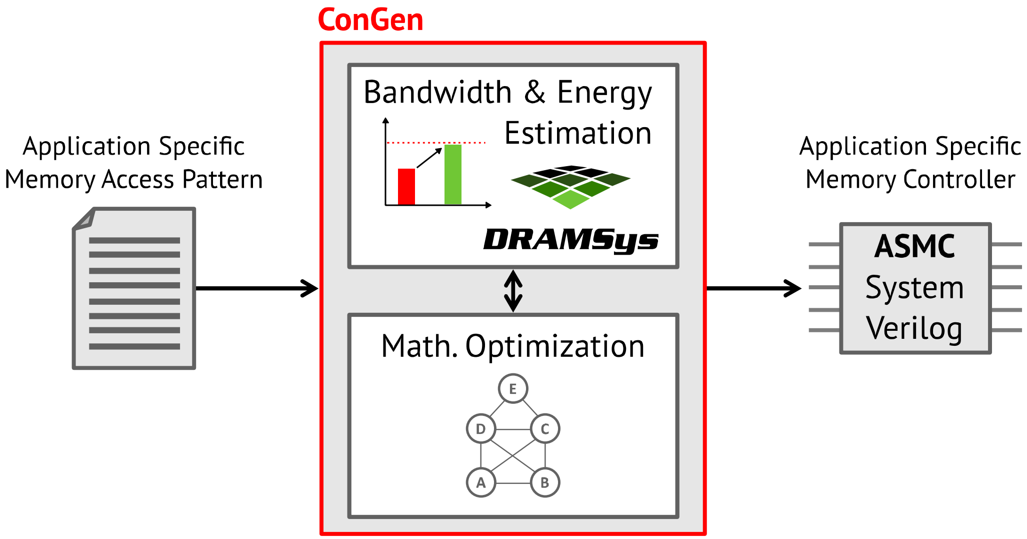
Any Bijective Mapping ← Application Knowledge

Output Address

ConGen Methodology

Exploit full application knowledge i.e. determinism of access pattern

- Minimize #row misses in same bank
- Decrease energy and latency, and increases bandwidth

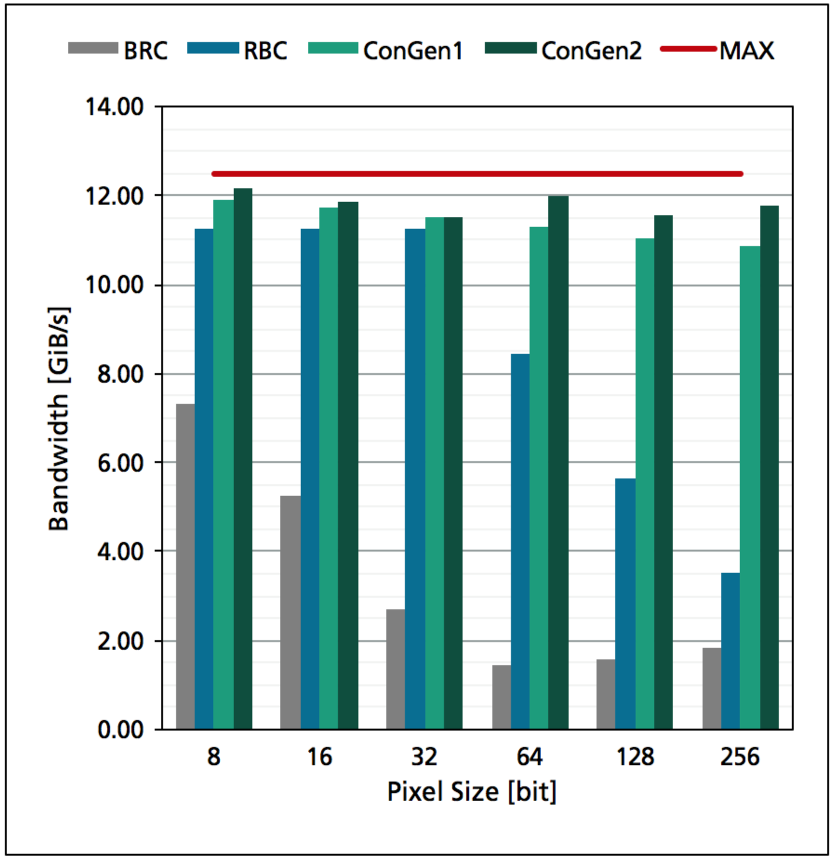


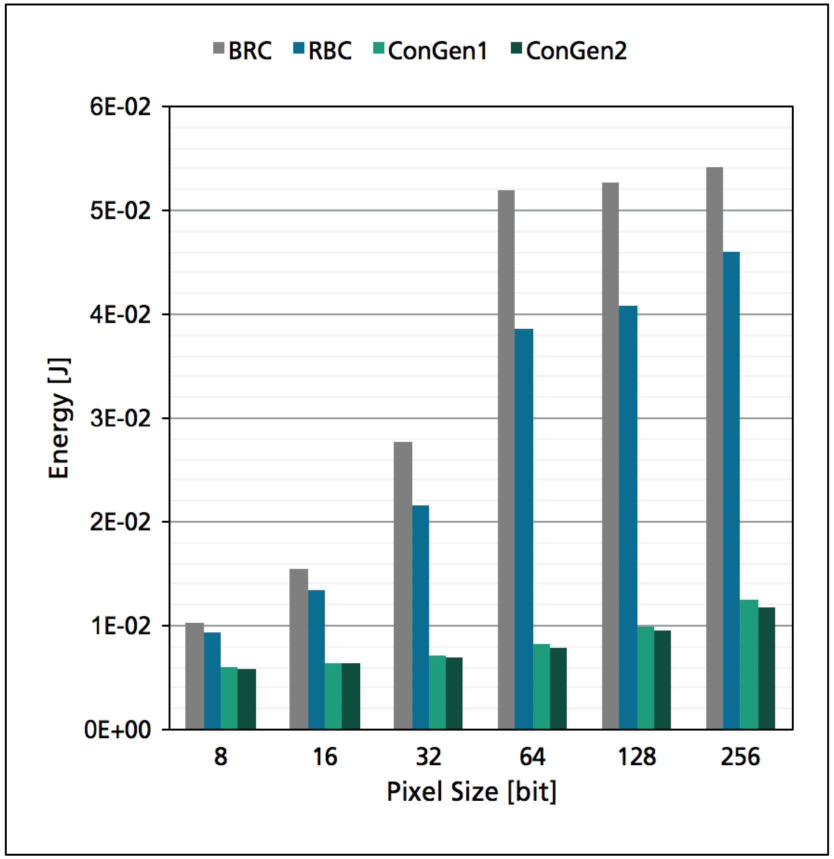
Optimization Problem

- Minimize number of row misses for all DRAM banks over an given logical memory access trace
- NP-hard problem

ConGen Methodology - Results

Industrial image processing task (Image Rotation 1024x 576 Pixel)

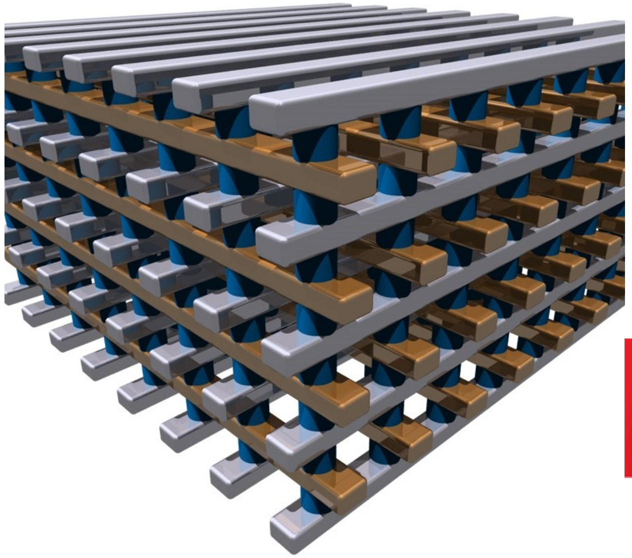




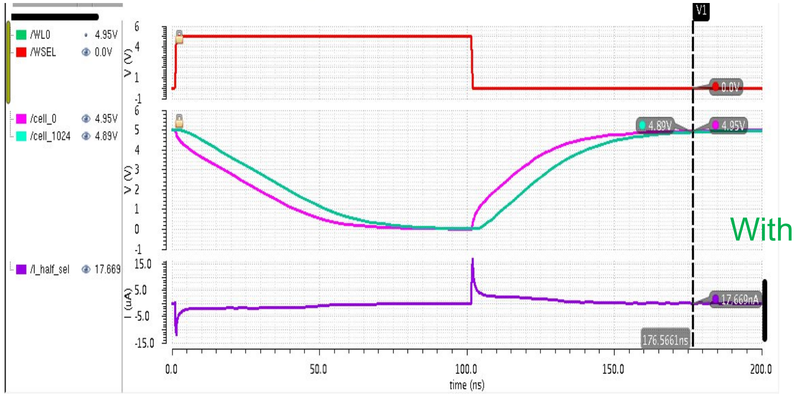
Legend: **BRC = Bank-Row-Column, RBC = Row-Bank-Column address mapping**

Heterogeneous Memory using ReRAM + DRAM

- Architecture level design space exploration tool to estimate the Timings, Area, and Power for high density ReRAM crossbar devices (**ReRAMSpec**)
- System Level (**SystemC**) and behavioral (**SystemVerilog**) modelling of ReRAM devices
- Circuit level modelling with **SPICE** of the ReRAM Array cross-section and periphery (Drivers, Sense-Amps etc.)

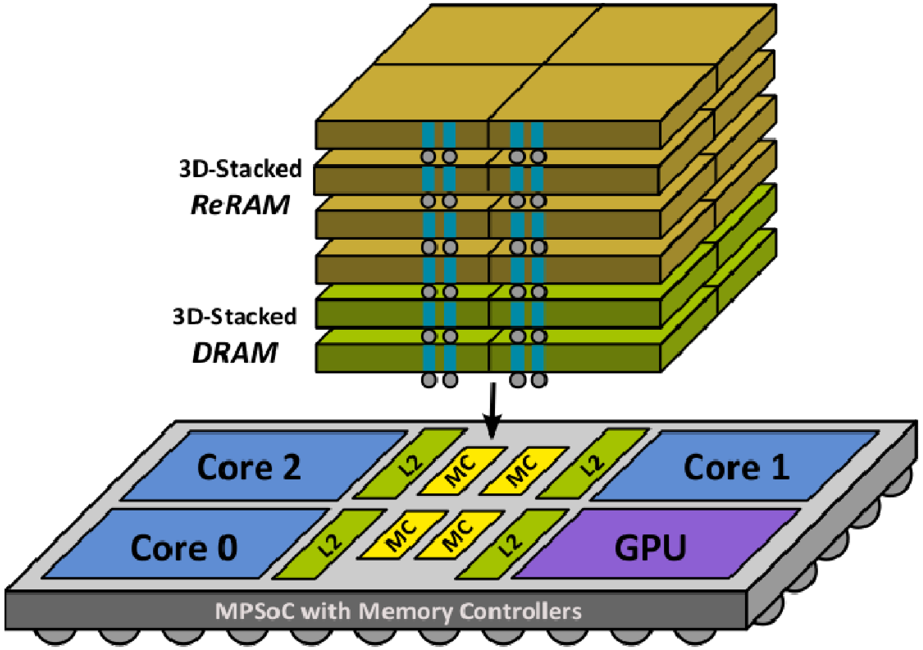


TECHNOLOGY RESEARCH INSTITUTE



With Selector

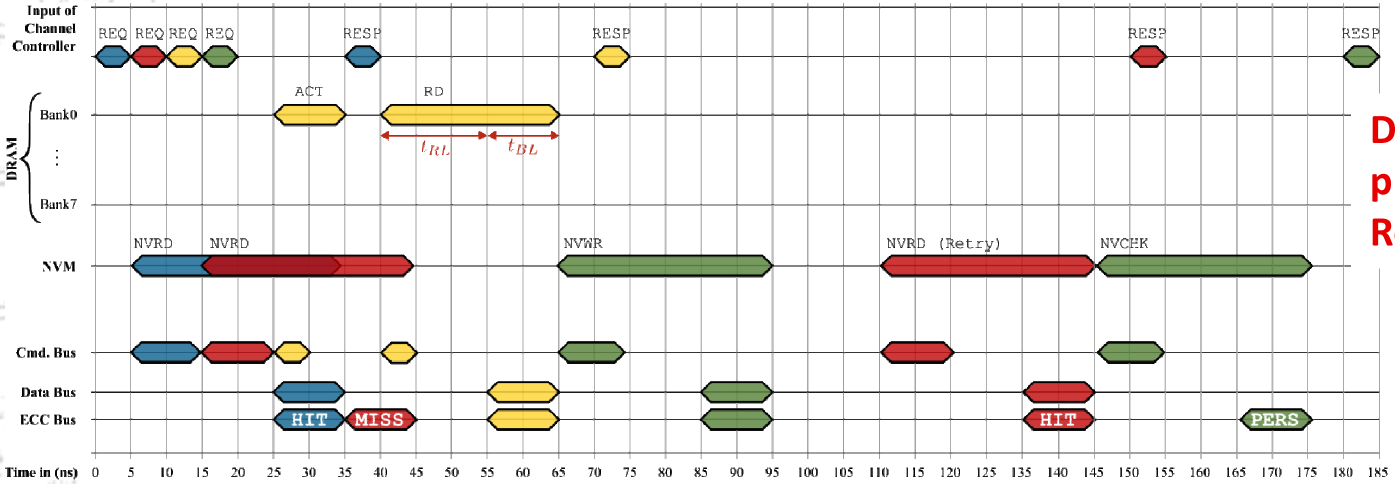
Heterogeneous 3D Memory System



- Each channel consists of DRAM (smaller capacity) and ReRAM (larger capacity)
- Special Memory Controllers (MCs) needed / Hybrid

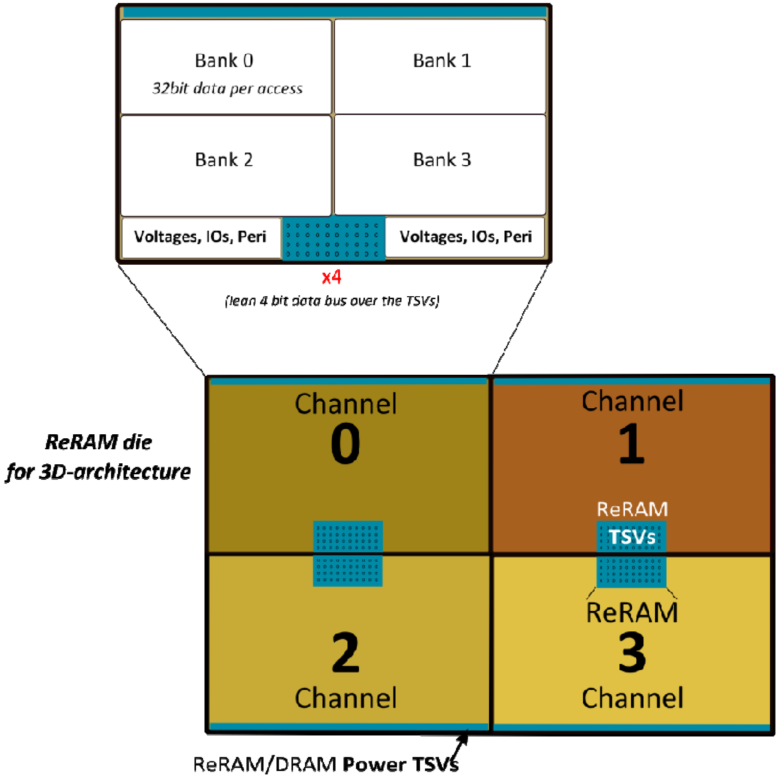
Two Options:

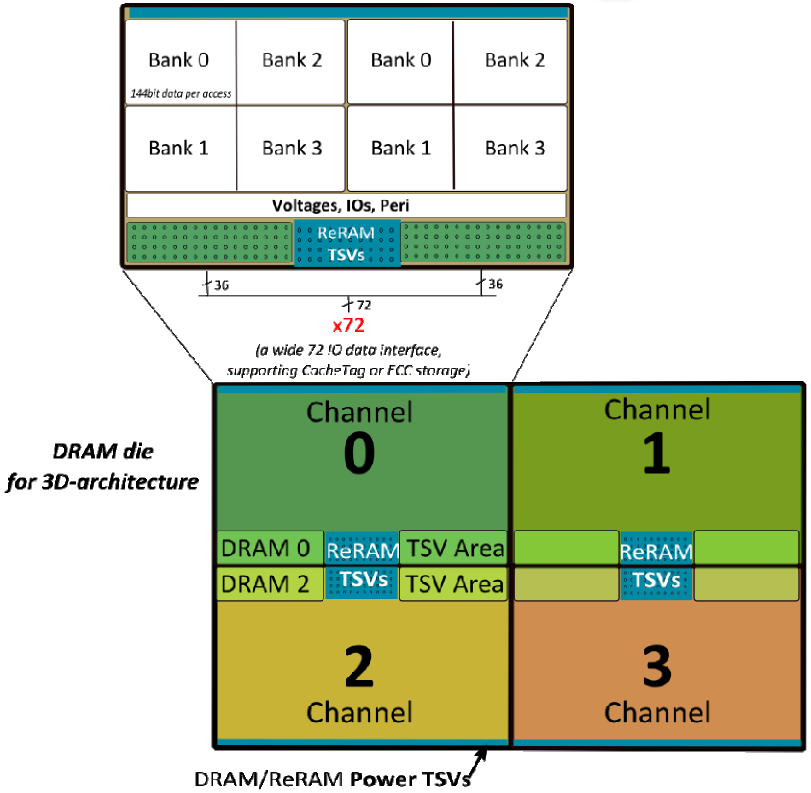
- DRAM as a Cache for ReRAM
- DRAM and ReRAM individually addressable



Definition of a new protocol for NVM (e.g. ReRAM) necessary!

Heterogeneous 3D Memory System





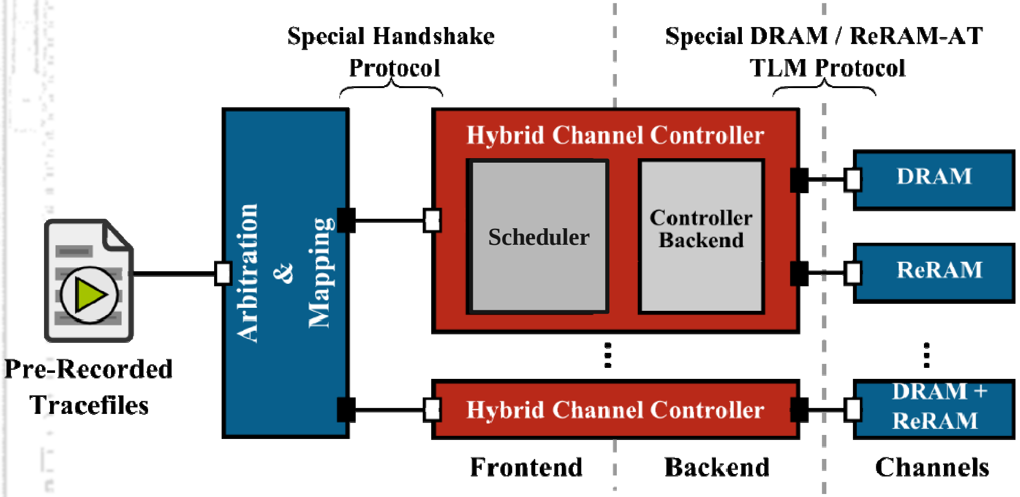
ReRAM Parameters for a Heterogeneous 3D Memory System

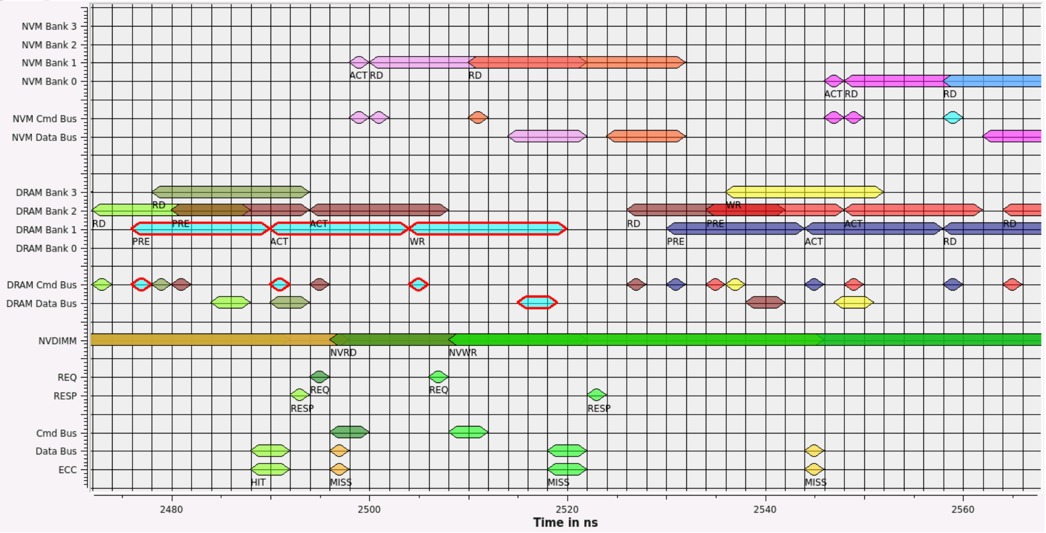
Capacity of the Die	16 Gbit
Die Size	11.7 x 12.6 mm
Number of Channels	4
Capacity/Channel and Layer (tier) size	4 Gbit
ReRAM/Logic Technology	28 nm
Number of Banks per Channel	4
IO width(s)	4 data lines
Interface and Frequency	DDR, Prefetch 8, 500 MHz
Maximum Bandwidth	1Gb/s per Pin

DRAM Parameters for a Heterogeneous 3D Memory System

Capacity of the Die	8 Gbit + 1 Gbit (for Tag or ECC)
Die Size	9 x 11.5 mm
Number of Channels	4
Capacity/Channel + Layer size	2 Gbit + 256 Mbit (for Tag or ECC)
DRAM Technology	22 nm
Number of Banks per Channel	4
Page size(s)	2K Bytes
IO width(s)	72 data lines
Interface and Frequency	DDR, Prefetch 4, 500 MHz
Maximum Bandwidth	1 Gb/s per Pin

Heterogeneous 3D Memory System





Trace player view of system simulation using our SystemC TLM 2.0 model

Option 1

- DRAM and ReRAM separate addressable

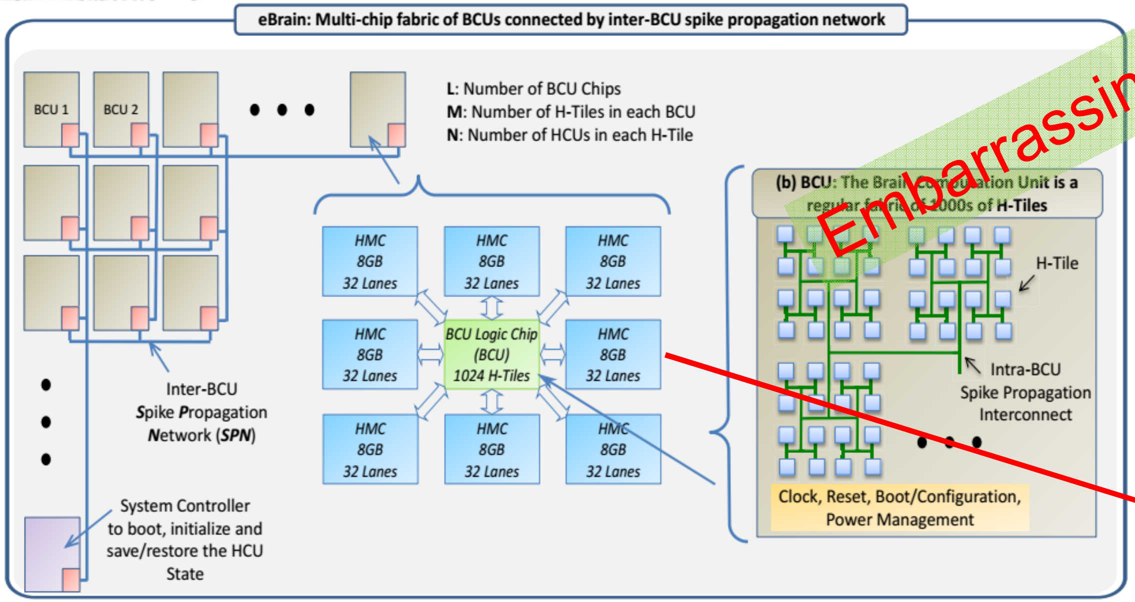
Option 2

- ReRAM is addressable
- DRAM is not directly addressable (Cache)
- Special Handshake Protocol (e.g. NVDIMM-P), to avoid unnecessary waits in case of Cache misses
- Special ReRAM protocol

Custom 3D-DRAM for eBRAIN II

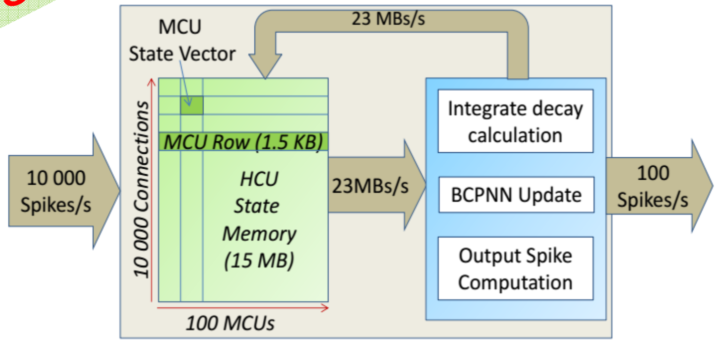
- A custom multi-chip design to simulate the human brain in real time using the spiking BCPNN (Bayesian Confidence Neural Network)
- The architecture for this algorithm is based on Hyper Columns Units (HCU) and Mini Columns units (MCU)
- The parallel computability of HCUs and MCUs makes this architecture hardware friendly
- Each HCU is an aggregation of 100 MCUs
- The hyper column unit has 10000 input connections and 100 output connections

Embarrassingly Parallel!



eBrain: Multi-chip fabric of BCUs connected by inter-BCU spike propagation network
BCU 1
BCU 2
L: Number of BCU Chips
M: Number of H-Tiles in each BCU
N: Number of HCUs in each H-Tile
HMC 8GB 32 Lanes
BCU Logic Chip (BCU) 1024 H-Tiles
Inter-BCU Spike Propagation Network (SPN)
System Controller to boot, initialize and save/restore the HCU State
(b) BCU: The Brain Computation Unit is a regular fabric of 1000s of H-Tiles
H-Tile
Intra-BCU Spike Propagation Interconnect
Clock, Reset, Boot/Configuration, Power Management

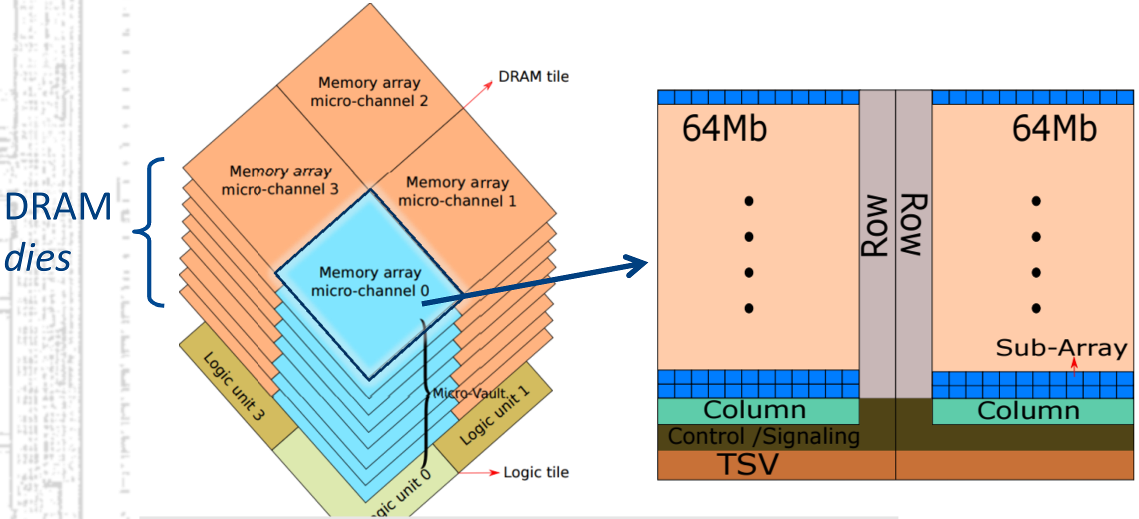
HCU updates:
Row-wise
Col-wise



HMCs consume ≈**40KW**

Custom 3D-DRAM for eBRAIN II

- Custom-optimized **3D-DRAM architecture** => 48 I/O DDR microChannel per HCU (1 – 2 mm^2 depending on the DRAM tech.) with 500MHz freq.
- Tailored **access** → using a technique called "**Row merge**", where we balanced the BW between Row-updates and Col-updates.



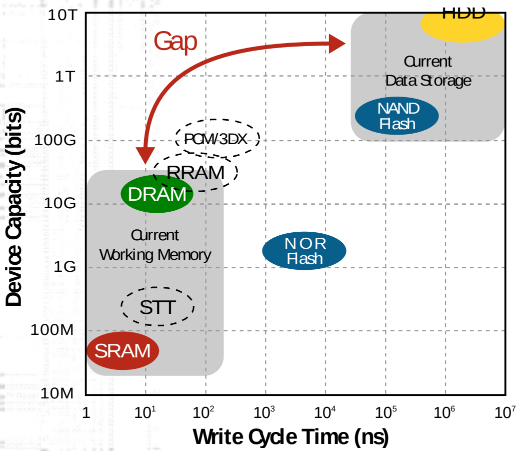
Species	# of HCUs	Average Power
Mouse	1.6×10^3	13 W
Rat	5.0×10^3	44 W
Cat	6.0×10^4	522 W
Macaque	2.0×10^5	1700 W
Human	2.0×10^6	**17 KW**

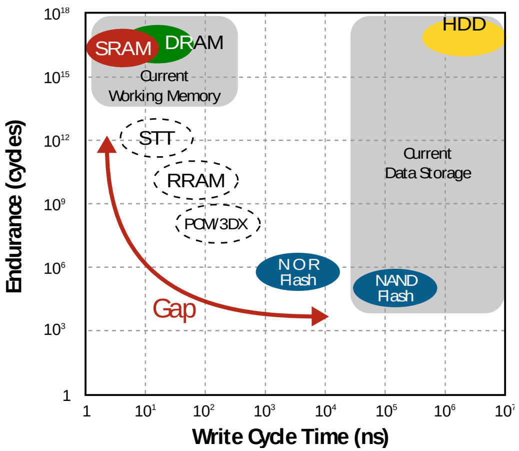
Matrix – Bank mapping of 4 HCUs:
→ *optimized data layout*

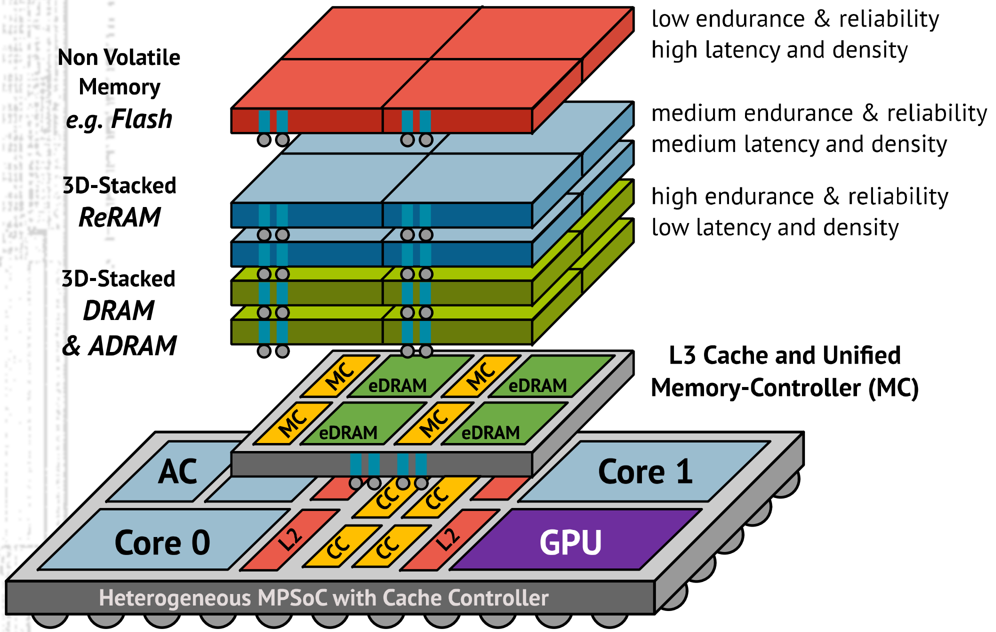
Bank 0	Bank 1	Bank 2	Bank 3
i cells HCU 2	i cells HCU 2	i cells HCU 3	i cells HCU 3
ij cells HCU 1	ij cells HCU 1	ij cells HCU 1	ij cells HCU 1
ij cells HCU 0	ij cells HCU 0	ij cells HCU 0	ij cells HCU 0

Bank 4	Bank 5	Bank 6	Bank 7
i cells HCU 0	i cells HCU 0	i cells HCU 1	i cells HCU 1
ij cells HCU 3	ij cells HCU 3	ij cells HCU 3	ij cells HCU 3
ij cells HCU 2	ij cells HCU 2	ij cells HCU 2	ij cells HCU 2

The Future is Hybrid/Heterogeneous






Non Volatile Memory *e.g. Flash*
3D-Stacked *ReRAM*
3D-Stacked *DRAM & ADRAM*
L3 Cache and Unified Memory-Controller (MC)
MC
eDRAM
MC
eDRAM
MC
eDRAM
MC
eDRAM
AC
Core 1
Core 0
CC
CC
CC
CC
L2
L2
GPU
Heterogeneous MPSoC with Cache Controller

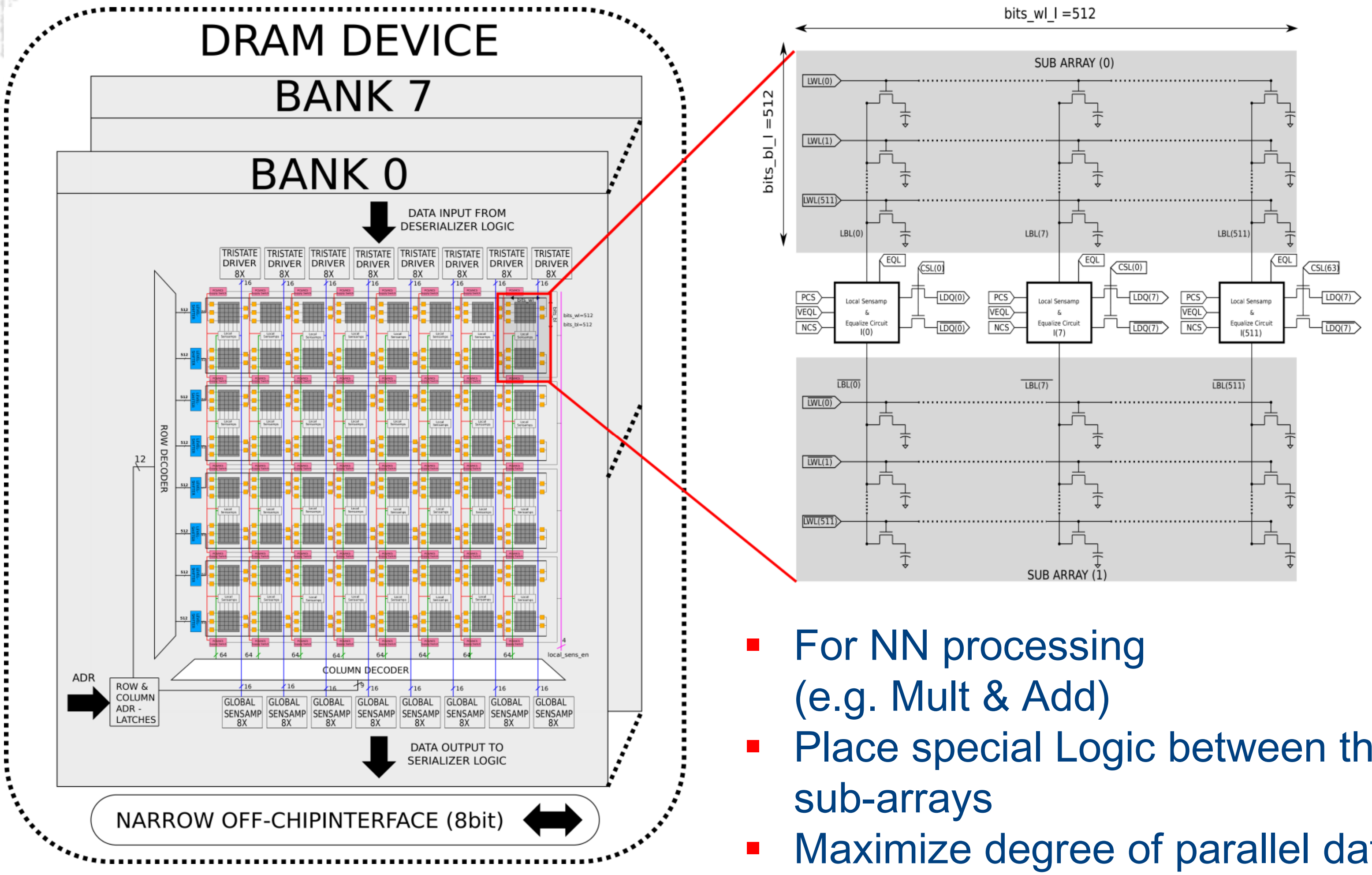
low endurance & reliability
high latency and density

medium endurance & reliability
medium latency and density

high endurance & reliability
low latency and density

- New memory technologies:
  - PCM
  - 3DXPoint
  - STT-MRAM
  - **ReRAM**
- DRAM **won't** be dead, but will change its role → maybe used as **Cache …**
- New memory **ECC** techniques
- Heterogeneous main memory systems:
  - **NVDIMM-P**
  - 3D MPSoCs / **3D Memory Stacks**
- New requirements on:
  - Compiler
  - OS
- **Processing in memory (PIM)**

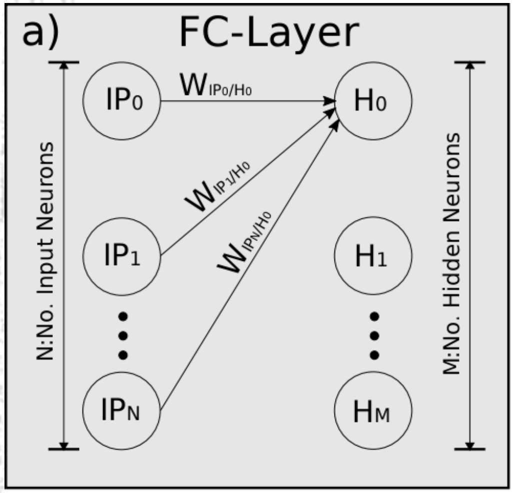
In/Near–Memory Processing


DRAM DEVICE
BANK 7
BANK 0
DATA INPUT FROM DESERIALIZER LOGIC
TRISTATE DRIVER 8X
ROW DECODER
12
ADR
ROW & COLUMN ADR - LATCHES
COLUMN DECODER
local_sens_en
GLOBAL SENSAMP 8X
DATA OUTPUT TO SERIALIZER LOGIC
NARROW OFF-CHIPINTERFACE (8bit)
bits_wl_l =512
bits_bl_l =512
SUB ARRAY (0)
LWL(0)
LWL(1)
LWL(511)
LBL(0)
LBL(7)
LBL(511)
EQL
CSL(0)
CSL(63)
PCS
VEQL
NCS
Local Sensamp & Equalize Circuit I(0)
Local Sensamp & Equalize Circuit I(7)
Local Sensamp & Equalize Circuit I(511)
LDQ(0)
LDQ(7)
SUB ARRAY (1)

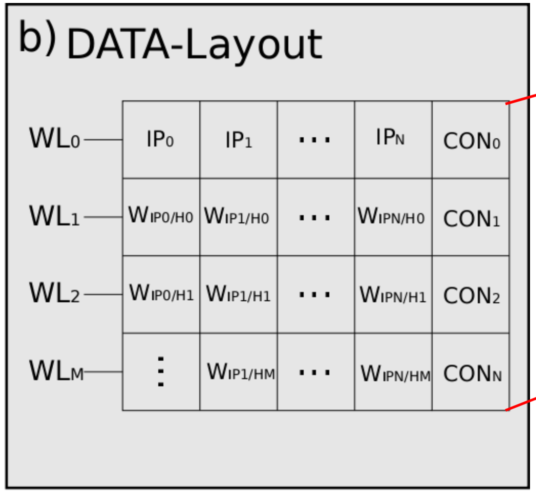
- For NN processing (e.g. Mult & Add)
- Place special Logic between the sub-arrays
- Maximize degree of parallel data processing e.g. 512/1024 bit in DDR3/4 devices

MICROELECTRONIC SYSTEMS DESIGN RESEARCH GROUP

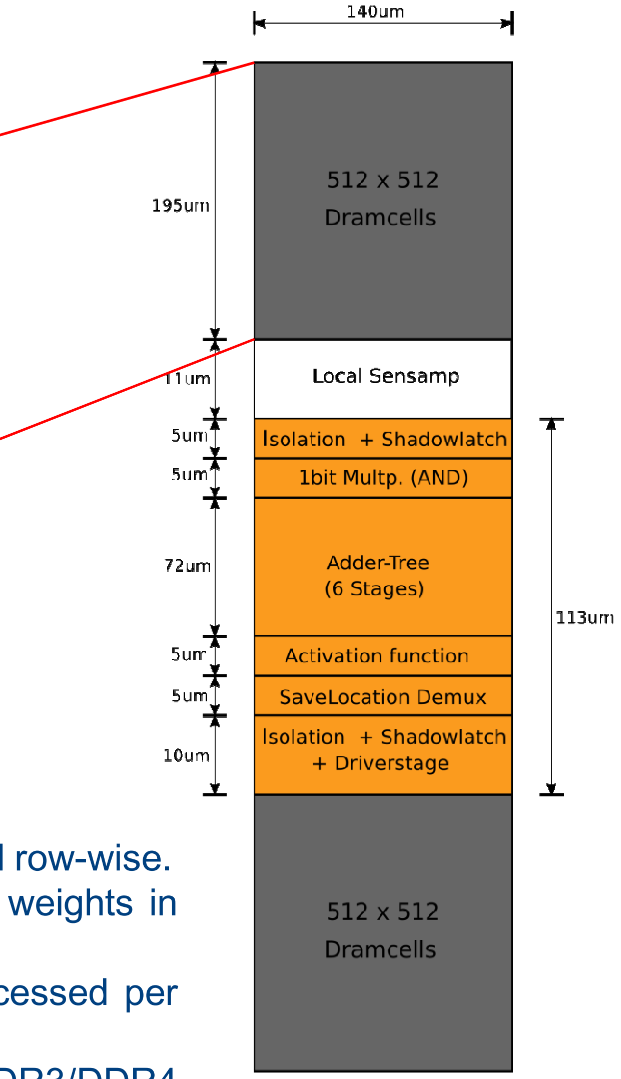
In/Near–Memory Processing



IP_x: Input Layer
$W_{x/y}$: Weight
H_y: Hidden/Output Layer
CON_x: Config Flag Layer Type
WL_x: Wordline (Mem. Row)


b) DATA-Layout

WL_0	IP_0	IP_1	⋯	IP_N	CON_0
WL_1	$W_{IP0/H0}$	$W_{IP1/H0}$	⋯	$W_{IPN/H0}$	CON_1
WL_2	$W_{IP0/H1}$	$W_{IP1/H1}$	⋯	$W_{IPN/H1}$	CON_2
WL_M	⋮	$W_{IP1/HM}$	⋯	$W_{IPN/HM}$	CON_N

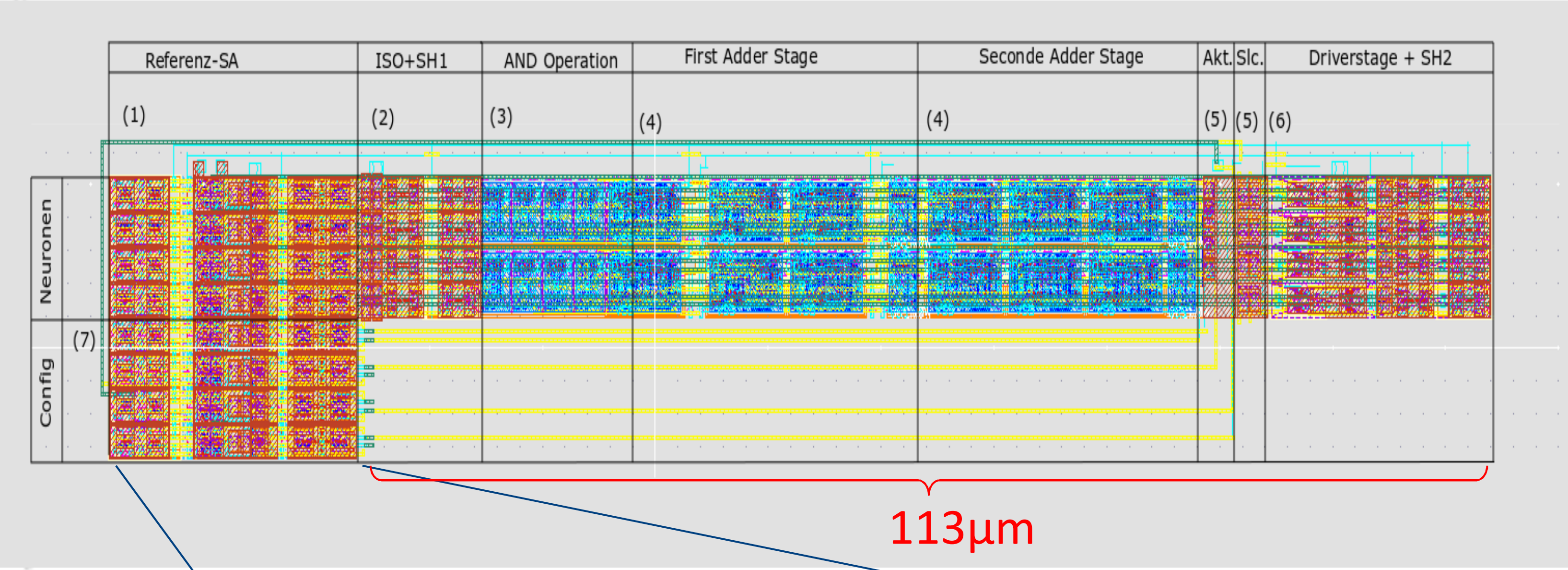


The data layout:

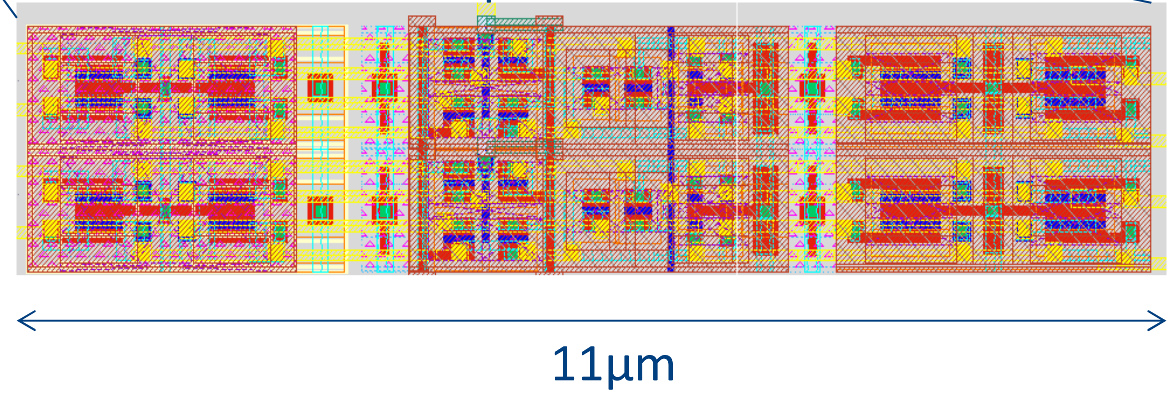
- Weights correspondent to separate neurons (H0, H1, ..) are stored row-wise.
- Process the inputs of a single neuron in parallel by reading the weights in parallel from the corresponding DRAM row
- Currently 100 values (100 bits because of 1-bit weights) are accessed per row per clock cycle
- Maximum bandwidth per clock cycle can be 512/1024 bits for DDR3/DDR4 minus the control bits!.

Layout Study – Feasibility

Neural Network Layout Implementation



Reference Sense-Amplifier



In/Near–Memory Processing

Current implementation:

Input data precision, QI [bits]	2
Weights precision, QW [bits]	1
Number of neurons per fully-connected layer, N	100
Number of layers, L	10
Access time per DRAM row, T [ns]	**20**
Number of sub-arrays per bank, S	16
Number of banks in DRAM device, B	8

Taking into account that the current implementation allows to access N weights per sub block in S sub-arrays in B banks of a single device in parallel, the computed throughput is:

N * 2[1] * S * B / T = **1.28 TOP/s**

[1] 2 comes from addition and multiplication considered as separate operations

Summary – Take-away messages

- Approximate DRAM and optimized Refresh control can be used to trade-off BW vs. reliability
- ConGen methodology to improve BW and energy
- Custom 3D-DRAMs have a large potential
- Hybrid/Heterogeneous architectures and Near/In-memory processing will be key

Thank you for Listening

For more information //ems.eit.uni-kl.de

For the tools

https://www.uni-kl.de/3d-dram/tools/